COMMODORE USERS

Ahoy!

48355 K

$2.50 /CAN.$2.75 SEPT. 1984



0 71896 48355 09

CARDCO "NOW" SOFTWARE

... available now for your Commodore-64™ and more!

A fine line of software developed by CARDCO for your Commodore-64 computer with all the features you should expect in much more costly software. CARDCO's "NOW" Series provides many unique and exclusive features and are packaged for easy reference, simple storage, instant recognition.

"WRITE NOW" ... WORD PROCESSOR SOFTWARE ... An excellent time saver, CARDCO offers the "Write Now" C/ Ø2 word processor program with built-in 80 column display. You see exactly what will print. All special codes can be transmitted to printers maintaining justification. Easy full-screen editing; works with any printer.

"MAIL NOW" ... MAILING LIST SOFTWARE ... CARDCO's D/ Ø1 "Mail Now" quickly (in memory) sorts by zip, category, name and state; fully compatible with "Write Now". Other fine features include: user-oriented; menu-driven operation; each disk supports 600 entries. Format can print single, double or triple labels across.

"SPELL NOW" ... Cardware D/ Ø4 ... a fine program designed as a spell checker for use with "Write Now" on the Commodore-64. A 34,000 word dictionary with two additional user constructed dictionaries. Menu-driven operation for ease of use. And "Spell Now" allows you to see each misspelled word in the context of your document for correction.

"FILE NOW" ... D/ Ø5 ... is a totally integrated, menu-driven database software package which interfaces with both the "Write Now!" for the 64 and the "Spell Now." 40K of working storage space is available with "File Now". "File Now" appears on the screen as index cards for easier manipulation of your data base; you see 5 index cards at a time. Cards are user defineable, i.e., user determines what goes where on the "index cards" and can sort by any given field. Every card has a general topic field which allows for quick sorting through cards.

"GRAPH NOW" INCLUDING ... "PAINT NOW" ... D/ Ø6 ... This disk-based graphic/logo generator is totally menu-driven. Allows for the development of pies, charts, bar graphs and other vivid graphic illustrations. Also has the ability to design, and print logos and high resolution pictures. "Commodore-ready"; interfaces with CARDCO'S "Write Now" Word Processor, "Mail Now", "Spell Now" and "File Now".

Write for illustrated literature and prices or see CARDCO Computer Accessories and Software wherever Computers are sold.



cardco, inc.

300 S. Topeka Wichita, Kansas 67202 (316) 267-6525

"The world's largest manufacturer of Commodore accessories."

Commodore™ is a registered trademark of Commodore Business Systems, Inc.

Reader Service No. 168

Ahoy!

CONTENTS

DEPARTMENTS

FEATURES

PROGRAMS

Cover illustration by James Regan

(Illustrations in screens by James Regan and Kevin Walter)

Publisher
Michael Schneider

Editor
David Allikas

Managing Editor
Robert J. Sodaro

Senior Editor
Tim Moriarty

Consulting Editors
Ben Bova
Morton Kevelson
Dale Rupert

Art Director
Raoul Tenazas

Associate Art Director
JoAnn Case

Art Production
Christopher Carter
Pauline Giordani
Eve Griffin

Technical Advisors
David Barron
Edward D. Laufer
Bob Lloret

Circulation Director
W. Charles Squires

Advertising Director
Lynne Dominick

Director of Promotion
Joyce K. Fuchs

Controller
Dan Tunick

Managing Director
Richard Stevens

Advertising Representative
JE Publishers Representative
6855 Santa Monica Blvd., Suite 200
Los Angeles, CA 90038
(213)467-2266
Boston (617)437-7628
Dallas (214)660-2253
New York (212)724-7767
Chicago (312)445-2489
Denver (303)595-4331
San Francisco (415)864-3252

AHOY! No. 9, September 1984. Published monthly by Ion International Inc., 45 W. 34th St., Suite 407, New York, NY, 10001. Subscription rate: 12 issues for $19.95, 24 issues for $37.95 (Canada $26.95 and $49.95 respectively). Application to mail at 2nd Class postage rates is pending at New York, NY and additional mailing offices.

Postmaster, send address changes to *Ahoy!*, 45 W. 34th Street, Suite 407, New York, NY 10001. All editorial and subscription inquiries and software and hardware to be reviewed should be sent to 45 W. 34th St., Suite 407, New York, NY 10001.

VIEW FROM THE BRIDGE

Have mercy, our readers begged! The programs in *Ahoy!* are so good that we can't resist typing them *all* in. . .but at 20-25 hours of typing per month, our fingers are going to succumb to bone disease! Put them on *disk* for us each month, they pleaded. We'll pay anything. *Anything!*

We agree—your fingers have earned a rest. Beginning this month, you can receive all the programs in each issue of *Ahoy!* on a verified disk—for only $7.95 (less if you subscribe). You'll find complete information on page 78 of this issue.

Included on our September disk (and in our September issue):

DOS by Bob Lloret, which reduces all the disk commands commonly executed by C-64 users to a single keystroke. (Turn to page 21.)

Salvage Diver by B.W. Behling, a C-64 game that takes you to the ocean floor in search of the treasure of the shipwrecked S.S. Marie. (Turn to page 22.)

The Castle of Darkness, part I of the text adventure presented this month and next in Orson Scott Card's *Creating Your Own Games on the VIC and 64.* (Turn to page 17.)

Sound Explorer, part of the conclusion of David Barron's *Unraveling the Mysteries of Sound,* which lets you have a party with the amazing capabilities of the C-64's SID chip. (Turn to page 29.)

Base Conversions by Drexel Gibson, providing 64 and VIC users with an easy means of switching from decimal to hexadecimal to binary. (Turn to page 77.)

Programs from back issues are also available on disk. See prices on the coupon on page 78. (Ultra-beginners please note: though each monthly disk will contain all the programs in a particular issue, you will only be able to run those designed for your system, be it 64 or VIC.)

As for the rest of the September issue of *Ahoy!:*

Sanjiva K. Nath's *In Search of a Word Processor* concludes this issue, with reviews of 10 top programs for the C-64. (Turn to page 11.)

On August's cover we promised *10 Uses for a Word Processor.* It was squeezed out at the last minute. Ed Hoornaert's informative feature appears in this issue, however. It really does. Honest. We're not kidding this time! (Turn to page 25.)

With the millions of new Commodore 64 owners, we felt many might not yet have purchased a disk drive or even a datasette. If that applies to you, we caught you just in time! You can save money with Anthony Scarpelli's *Building a Cassette Interface for your C-64.* (Turn to page 31.)

Dale Rupert takes a peek at the memory locations associated with the video display in this month's *Rupert Report* on *Video RAM.* (Turn to page 70.) Of course, Dale has a new batch of *Commodares* waiting for you as well. (Turn to page 47.)

Our *Scuttlebutt* and *Reviews* sections have been shortened this time by space restrictions. They'll be back to full scale next month, along with *Flotsam.* (Please keep writing as you have been—either to chat or to ask technical questions. David Barron will be answering as many of the latter as space permits in our newest column—*S.O.S.*—premiering shortly.)

Thanks for picking up the September *Ahoy!*

—David Allikas



Commodore 64™ Owners, Relax...

with Mirage Concepts software

Mirage Concepts has mastered the art of uncomplicating software. Before you buy—we help you determine which Mirage Concepts package will meet your need. No guesswork! With your purchase comes a menu-driven program ranked by independent evaluators nationwide as among the finest available. Relax as you learn how to operate your program with clear, concise tutorials written by professional writers... not programmers. For consultation on your special questions, technical support personnel are standing by on a toll-free basis.

For Brochures, Support and Information, Call...
(800) 641-1441
In California, Call...
(800) 641-1442

DATABASE MANAGER, $89.95
• 100% Machine Language • Free Form Design • Sort On Any Field • Calculated Fields
• Interfaces to W.P. • Record Size = 2,000 Characters

ADVANCED REPORT GENERATOR, $49.95
• Companion to Database • Totals and Subtotals • Field Matching • Expanded Reports
• Sorting (Up & Down) • Calculated Fields

WORD PROCESSOR, Professional Version $89.95
• 80 Col w/o Addt'l Hdwr • 100% Machine Language • Spelling Checker (30,000 Words)
• Over 70 Single Keystroke Commands • Printer Command File • Interfaces to Database

WORD PROCESSOR, Personal Version $39.95
• 100% Machine Language • True Word Wrap • Printed page/line/character counters
• Right Justify, Center • Printer Command File • Interfaces to Database

MIRAGE CONCEPTS, INC.
2519 W. Shaw Ave., #106 • Fresno, CA 93711
TM—Commodore 64 is a Registered Trade Mark of Commodore Electronics, Ltd.

Reader Service No. 169

IT STARTED IN '81 when our president, the designer of the VIC-20, left Commodore to open his own company.

His goal was to build an American tradition ... THE BETTER WAY.



MSD SYSTEMS, INC... WE BUILD IT BETTER!



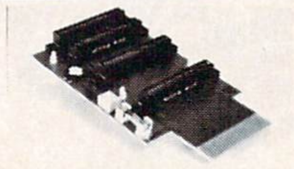
Our **RS-232 Serial Interfaces** allow you to connect printers, modems, plotters, and other input/output devices to your 64 or VIC-20.



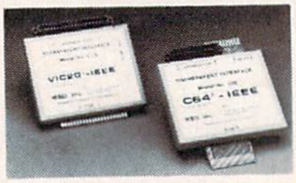
A parallel interface allows direct access between a centronics printer and your software program, saving you loading and waiting time. Our **CPI** is compatible with most software written for the 64 and VIC-20.



For professional-quality video and audio output, our **Monitor Link Cables** can connect your 64 or VIC-20 to a high-resolution black/white or color monitor, or a CRT terminal, and to a stereo system.



The **CEX-4** Expandoport gives you four additional ports for interfaces and peripherals.



Save time and hassle with the greater speed of an **IEEE Interface.** The unit is 'transparent' to your computer's expansion port, allowing hookup to any peripheral without interference. With this interface, your 64 or VIC-20 can gain CBM/PET-type control over the IEEE-488 bus.

Commodore produced good computers that were economical for families. Our president wanted to provide Commodore owners with better peripheral products.

In less than two years, MSD was shipping interfaces that expanded the Commodore 64, VIC-20, and PET computers into business and developmental applications ... THE SERIOUS WORLD OF COMPUTING.

Next, determined to BUILD IT BETTER, MSD developed SUPER DISK I and II ... exceptionally fast and durable disk drives that NEVER OVERHEAT.

MSD's dual drive formats, copies, and verifies in less than two minutes ... compared to 30 or 40 minutes with TWO Commodore 1541s. This unparalleled speed has made SUPER DISK II the hottest product introduced for the Commodore line of computers!

Add the six month warranty, and you can see why our dealers and their customers believe in us. MSD SYSTEMS, INC. ... Call us today for your nearest local dealer.

Dealer and distributor inquiries invited.

MSD SYSTEMS, INC.

10031 Monroe Dr. • Suite 206 • Dallas, Texas 75229
214-357-4434 • Outside Texas 800-527-5285

Reader Service No. 189

SCUTTLEBUTT

CONSUMER ELECTRONICS SHOW REPORT • COMMODORE 16 AND PLUS/4 COMPUTERS • WINDHAM CLASSICS • MATH, SCIENCE, AND GRAMMAR PROGRAMS • LOW-COST KEYBOARDS • MUSIC SYNTHESIS SOFTWARE • GRAPHICS PACKAGES • NEW GAMES FROM DATASOFT AND ACTIVISION

THE HITS KEEP COMING

At last January's Consumer Electronics Show (the semiannual showcase for all that's new in the industry), Commodore towered over its competitors as the clear microcomputing leader. This June's exhibition only reaffirmed Commodore's position, with more manufacturers than ever falling over one another to stoke the C-64 with software and peripherals. An occasional booth still displayed software only for the Atari or Apple—but we're betting that we'll see them at next January's show with Commodore added, or we won't see them at all.

Commodore 64 owners should be pleased to know that their computers will continue to be supported by a larger array of software than any other micro. The heavy competition should also lead to lower prices. Some manufacturers at CES showcased C-64 programs for as little as $10.00—and in one case $5.00.

The prognosis is not as good for VIC 20 owners. Support for that machine has long since peaked and is heading steadily down. Commodore will not confirm that production of the VIC has ceased/ will cease; the company line is that as long as demand for the computers continues, Commodore will continue manufacturing them. But that demand, based solely on the VIC's lower price as compared to the 64, will drop off in the face of the just-as-cheap and significantly more powerful. . .



The Commodore 16 is designed to introduce the novice to computer literacy.
READER SERVICE NO. 170

COMMODORE 16

As predicted in the May *Ahoy!*, reiterated last issue, and finally verified at the June CES, Commodore will manufacture a 16K computer designed to offer the first-time computer user an inexpensive (around $100) introduction to microcomputing. Built-in features of the Commodore 16 (to be advertised as "The Learning Machine") will include machine language monitor, graphics and sound commands, BASIC 3.5, and screen window capability. It will work with the MPS 802 dot matrix printer.

If the description of the C-16 sounds familiar, it may be because the new machine is basically a Commodore 264 with less memory and some other shortcomings. The latter computer, announced in these pages as early as March, was also exhibited at the June CES—as it was at the January CES. While still unavailable, it sported a new name: the Plus/4. (Commodore felt the name change necessary to prevent the mistaken assumption that the new machine has four times the memory of the C-64.) It will feature not one, as Commodore previously announced, but four built-in programs: word processing, database, spreadsheet, and graphics. Price will be around $300.

The Plus/4 is scheduled for early fall release, the C-16 for sometime in the second half of 1984.

Commodore, 1200 Wilson Drive, West Chester, PA 19380 (phone: 215-431-9100).

CLASSIC SOFTWARE

Richard Herring said it in our Educational Software series: anyone who thinks educational software discourages kids from reading isn't playing with a full circuit board. Further debunking the theory comes the Windham Classics series from Spinnaker, a line of graphics/text adventures based on enduring literary works.

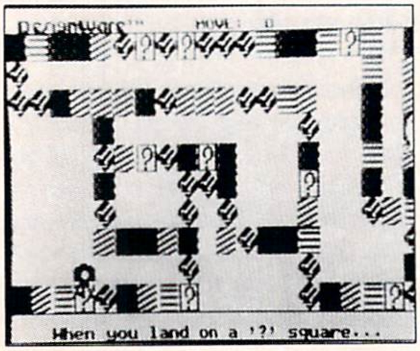
For players aged 10 to adult, the games place the player in the role

of the novel's hero or heroine, facing the problems they face and compelled to make decisions. Available in October on disk for the C-64 will be *Swiss Family Robinson, The Wizard of Oz, Gulliver's Travels,* and *Below the Root* (based on the Green Sky trilogy). Following those will be *Treasure Island* and *The Wind in the Willows.* Prices will be in the $30-$40 range.

Windham Classics, Spinnaker Software Corp., One Kendall Square, Cambridge, MA 02139 (phone: 617-494-1200).

EDUCATION PROGRAMS

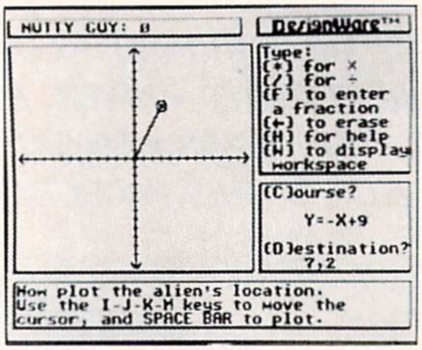
Three educational programs from DesignWare in addition to the *Notable Phantom* package described elsewhere:



Grammar Examiner, *C-64 edition.*
READER SERVICE NO. 171

The Grammar Examiner, a computerized board game, starts kids 10-14 off as cub reporters and lets them work their way up to editor-in-chief by correcting copy and answering grammar questions. Included are over 150 multiple choice questions and paragraphs with numerous grammatical mistakes; additionally, the built-in grammar editor lets you add your own. Available in August.

In order to re-create the course of his lost sister ship, the 13-18 year old player of *Mission: Algebra* must solve problems that involve coordinating pairs on a



Mission: Algebra—*outer space plot.*
READER SERVICE NO. 173

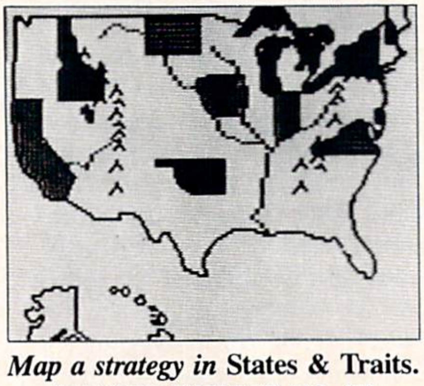
graph, determining the equation of a line, and solving for "x" and "y" coordinate pairs. Touching the help key will cause the base station to respond with two levels of strategies for solving the problem. Scheduled for September or October release.

States & Traits, for families with children aged nine up, requires players to place states and topographic features in their proper positions on a U.S. map and answer questions about U.S. landmarks and history. Already available.

Each on disk for the C-64, at $44.95.

DesignWare, 185 Berry Street, San Francisco, CA 94107 (phone: 415-546-1866).

First Star Software touts its *U.S. Adventure* as the first true educational program to utilize adventure game techniques. The player joysticks cross-country, entering dif-



***Map a strategy in* States & Traits.**
READER SERVICE NO. 172

ferent states and scoring points by answering questions about them.

Also from First Star: the first installment in its Romper Room Little Learner series, employing such characters as Do-Bee, Up Up, and Kimble from the TV series to teach young children a variety of skills. This first, *Romper Room's I Love My Alphabet,* uses an animated character that dances, hides, jumps, etc., to reinforce the meanings of action words.

Both for the 64 from First Star Software, Inc., 22 East 41st Street, New York, NY 10017 (phone: 212-532-4666).

Because computers operate in a linear fashion, most educational software does likewise, claims Prentice-Hall. But the human brain operates in parallel fashion, touching on a variety of subjects at once. So the Arrakis Advantage series they will begin distributing in time for the fall '84 semester will utilize the Socrates Learning Environment to allow students to interrupt the program at any point to ask questions, test themselves, review, and explore other subjects. Socrates himself leads students through the programs which cover a variety of math and science subjects.

Coming in August: *Algebra I, Volumes 1* and *2; Algebra II, Volume 1;* and volumes 1 of *Chemistry, Geometry,* and *Physics.* In September: *Algebra III, Volumes 1* and 2 and volumes 2 of *Algebra II, Biology, Chemistry,* and *Physics.* In October: *Biology, Volumes 1, 3,* and *4, Geometry, Volume 2,* and *Statistics, Volumes 1* and 2. $39.95 each, on disk for the C-64.

Prentice-Hall, P.O. Box 819, Englewood Cliffs, NJ 07632 (phone: 201-592-2611).

I PROGRAM THE SONGS

In the beginning, C-64 owners

played music right on their keyboards. Soon after that came interfaceable piano-style keyboards that brought users much closer to an actual musical experience—and the bottoms of their bank accounts. Now, combining low price and a measure of realism, come piano-style keyboards that fit over your 64's keys.

DesignWare's Notable Phantom music education game ($49.95) includes a polyethylene keyboard overlay providing an octave and a half of black and white "keys." The software, first in DesignWare's Musical Teacher series, uses ghouls and ghosts to lead children aged 5-10 through a variety of exercises teaching keyboard and notereading skills.

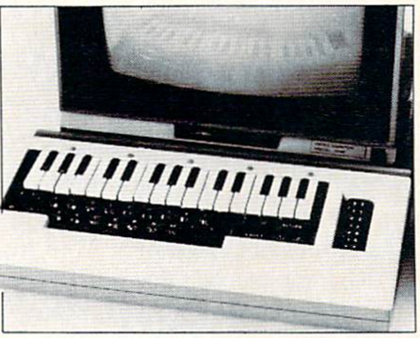
DesignWare, 185 Berry Street, San Francisco, CA 94107 (phone: 415-546-1866).

The 31 plastic keys of Sight & Sound Music Software's Incredible Music Keyboard ($39.95) let you play over an eight-octave range. Using the included music software you can create piano, guitar, synthesizer, and other musical sounds. If you like, you can start playing music almost instantly, using the included ABC note stickers and Letter Music Song Book. Also included is the Melody Chord Song Book for the more experienced. Software available from Sight & Sound for use with the keyboard includes the Kawasaki Synthesizer, Kawasaki Rhythm Rocker, the Music Processor, 3001 Space Odyssey, Tune Trivia, and Rock Concert.

Sight & Sound Music Software, Inc., 3200 South 166th Street, New Berlin, WI 53151 (phone: 414-784-5850).

UPDATES FROM ENTECH

One more musical note: EnTech has revised its *Studio 64* program to incorporate high-res graphics



Incredible Music Keyboard: 31 keys. READER SERVICE NO. 174

that make the notes easier to read, full musical notation including sharps, flats, and ties, and the capability of using control keys to move from one voice to another, adjust ADSR envelopes and filters, and save and load songs more quickly. Four new original sample songs are also included. Price is $39.95, or mail your old version plus $10.00. 10-song demo disk is available for $5.00 (no postage charge if prepaid).

EnTech also announced a "talking" version of its Space Math educational program that will reproduce the intonations, accents, and character of real speech. Similar enhancements of Management System 64 and Studio 64 (educational version) will follow later this year. Price of Space Math is $39.95.

EnTech Software, 10733 Chiquita, Studio City, CA 91604 (phone: 818-768-6646).

NEW GAMES UPDATE

Activision's *Pitfall II: Lost Caverns,* forecast in these pages in July, is now available. This sequel takes Pitfall Harry (plus niece Rhonda and mountain cat Quickclaw) to Peru in quest of the stolen Raj Diamond. The caverns consist of two cliffs beveled by ledges and separated by chasms (which Harry crosses with a balloon), and a river with waterfalls.

Unsuccessful encounters with scorpions, bats, condors, and other deadly pests send Harry back to the last mystical healing cross he touched. C-64 disk is $31.95, cartridge $34.95.

Four more C-64 games from Activision, slated for fall release:

As the *Zone Ranger* you must brave an unexplored sector of the universe, surviving deadly drones, planetoids—and battle on thirty graduated levels. Passing through the Super Portal puts you in a space vacuum from which you must steal elements to replenish your power.

In *Explorer* you probe the ruins of a deserted planet, selecting defense systems, reading complex maps, and crossing danger-laden plains in a hunt for artifacts, tools, and other items needed for survival.

Wonderbolt puts you in the overpaid shoes of a construction worker, racing the clock to bolt uncooperative girders into their blue-print pattern.

To complete our descent from the exotic to the mundane, *Camp Clean-Up* lets you, as a park ranger, pick up litter from the campgrounds and lake, rescue swimmers, and avoid other camp dangers.

Activision, Inc., 2350 Bayshore Frontage Road, Mountain View, CA 94043 (phone: 415-960-0410).

Four C-64 games from Datasoft, each retailing for $29.95:

Trying to keep your *Mancopter* on its flight pattern requires keeping flocks of birds out of your propellers and dodging opponents that can land you in shark- and squid-infested waters.

Meridian III requires you to guide your spaceship past energy globes and space mines and through the Warp Tunnel en route to assaulting the Dragon Lord's fortress city.

Juno First, wherein you must destroy waves of aliens encircling your spaceship, is one of two arcade adaptations. The other, *Lost Tomb*, lowers you into 91-chamber maze within an ancient Egyptian tomb packed creatures.

Datasoft, Inc., 19808 Nordhoff Place, Chatsworth, CA 91311 (phone: 818-701-5161).

TELECOMMUNICATIONS PROGRAM

Though it sounds like a fast food restaurant item, *McTerm 64* is actually a communications program for use with a C-64, disk drive, and modem. It allows you to control communications settings, onscreen characters, and transmission speed (up to 1200 baud). Other features include a clock to help keep track of that long distance bill, word wrap and auto linefeed options, and the ability to preset the program to send files at a desired time. Suggested retail $49.95.

Madison Computer, 1825 Monroe Street, Madison, WI 53711 (phone: 608-255-5552).

LET'S GET GRAPHIC

The Activision Pencil is named for its goal of reducing graphics programming to the simplest possible terms. The user can draw by using keyboard commands or entirely via joystick. The first screen, a computer "palette," includes over 75 graphics and musical commands; on the second a pencil (complete with eraser) executes the commands. On disk and cartridge for the C-64, in the fall.

Activision, Inc., 2350 Bayshore Frontage Road, Mountain View, CA 94043 (phone: 415-960-0410).



Peripheral Vision ***has 35 textures.***
READER SERVICE NO. 175

Peripheral Vision, available with Futurehouse's Edumate Light Pen ($59.95 the set) or separately, offers 15 colors, 6 brush widths, 35 textures, mirror, zoom, copy, fill, and move, and the ability to mix keyboard characters with graphics. For the C-64.

Futurehouse, P.O. Box 3470, Chapel Hill, North Carolina 27514 (phone: 919-967-0861). □

IN SEARCH OF... A WORD PROCESSOR FOR YOUR COMMODORE 64 PART II: REVIEWS

By Sanjiva K. Nath

Last issue we introduced you to word processing on the C-64 and its applications. We also discussed the features associated with the commercially available word processing programs. Using these features as guidelines, a set of criteria were established to assist you in selecting the right program to match your specific needs.

In this article, we will provide brief reviews of ten popular word processors for the C-64. (These reviews have been adapted from *The Commodore 64 Software Buyer's Guide* by Sanjiva K. Nath and Terry Silveria—Robert J. Brady Co.) Along with our previously established criteria, these reviews will provide you with tools to compare and evaluate any given word processing program from dozens that are presently available for the C-64. Following these reviews, we have also provided a table that compares the features in these programs. For more details on the specific features, refer to the glossary provided last issue.

EASY SCRIPT

Commodore; diskette, $49.95

This is a comprehensive word processing program available at an exceptionally low cost. It can be used with disk or tape, and any properly interfaced printer. It may be configured from 40 to 240 columns for text entering. This is especially useful when working with large tables. DOS functions are available from within the program. The main memory can hold up to 799 lines of text. The command line on top of the screen lets you know what function or mode you are currently working in. This line is also used by the system to display any prompts for additional information as well as DOS error messages.

The program supports the standard editing keys on the C-64 for convenience in text entering and editing. Output may be directed to the screen prior to printing, so that you may look at the format before printing. You can also send special control codes to your printer to access special print effects. Files may be linked on disk for continuous printing.

Special features include the 40-240 column text entering, decimal tab set, capital lock, and automatic horizontal and vertical scrolling (panning). Mail merge facility is also available for creating personalized form letters. The program works with the *Easy Spell* spelling checker, also from Commodore.

Commodore Business Machines, 1200 Wilson Drive, West Chester, PA 19380 (phone: 215-431-9100).

HESWRITER

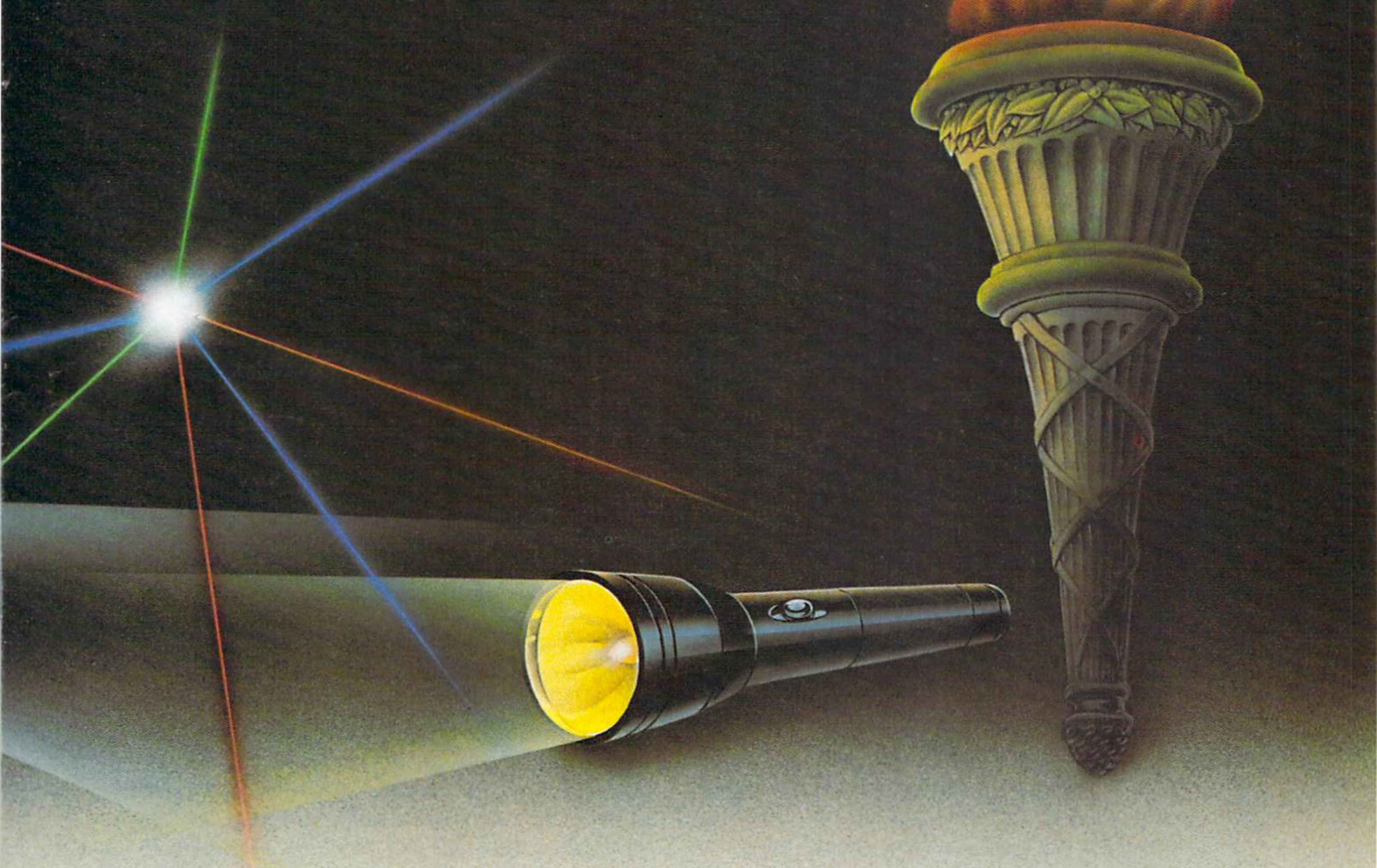
HesWare; cartridge, $44.95

Heswriter is a simple, easy-to-use word processor for the C-64 which is directed at the novice user. However, it lacks some sophisticated features found in other comparably priced word processors, the most important being the lack of full-featured editing. The program is cartridge-based, with no printer configuration or DOS functions support.

A special text entering mode is used to enter text. This mode may not be used to fully edit text. 27 screen lines of text (36 characters per line) are allowed with no word wraparound feature. The edit mode allows text editing, although in this mode you cannot enter text. Each line of text is identified by the text editor with a line number, which may be

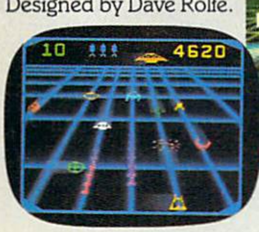
FOR YOUR COMMODORE 64.

DIFFERENT LIGHT.



As you suit up you see the webbed forcefield surrounding your planet. Holding it. Trapped with no escape. No hope. Except you: The Beamrider. The freedom of millions depends on you. Alone you speed along the grid of beams that strangle your planet. You must destroy the grid sector by sector. Your skills and your reflexes alone will determine the future of your people. Take their future in your hands. Designed by Dave Rolfe.



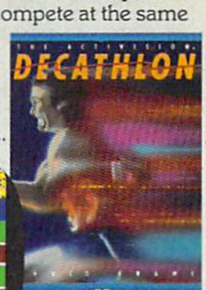


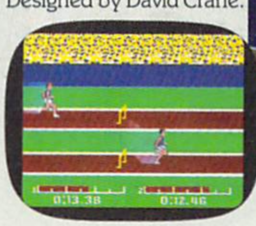
You can almost hear the quiet. And it's your job to keep it that way. A toy factory at midnight. Did you hear something? Guess not. Wrong! Suddenly balloon valves open, conveyor belts move and a whole factory full of toys goes wild. Even the robot, their latest development, is on the loose and after you. Capture the runaway toys. Restore order. Restore peace. Restore quiet. Do something! Hurry! Designed by Mark Turmell.





You made it. The Olympics. You hear languages you've never heard. And the universal roar of the crowd. You will run. Hurl. Vault. Jump. Ten events. One chance. You will push yourself this time. Further than ever. Harder than ever. But then . . . so will everyone. The competition increases, now two can compete at the same time. The crowd quiets. The starting gun sounds. A blur of adrenalin. Let the games begin. Designed by David Crane.





ACTIVISION.

We put you in the game.

Reader Service No. 167



Covered in detail in our March issue, **Easy Script** *offers comprehensive word processing at an exceptionally low cost.*
READER SERVICE NO. 132

used to find that line for future editing. Text is printed by specifying the beginning and ending line numbers. Control characters are placed within the text for formatting an output.

A great deal of effort is required to use this word processor, and for the novice user, it may prove to be a useful learning experience.

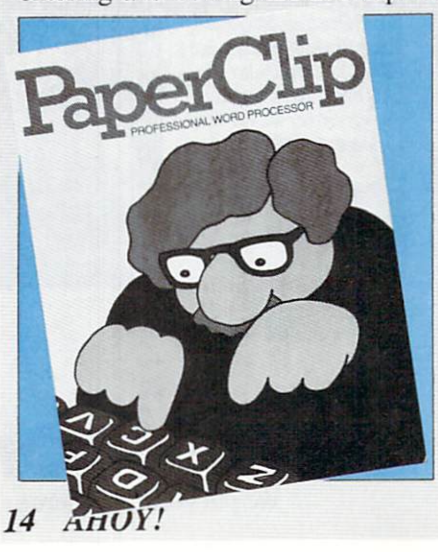
HesWare, 150 North Hill Drive, Brisbane, CA 94055 (phone: 415-468-4111).

PAPERCLIP

Batteries included; diskette, $125.00

PaperClip is one of the most comprehensive word processing programs for the C-64. It offers most standard editing and print formatting features along with a few unusual ones not encountered elsewhere. Considering the price tag, decide whether these novelties are worthwhile to you.

The program is configurable with a number of printers via the special printer files available on the system diskette. DOS function support and screen color settings are available in the initial options. Text entering and editing are accomplished using the standard Commodore screen editor keys and functions. A command line at the top of the screen lets you know what function or mode you are working in. This line is also used by the system to display any prompts for additional information as well as DOS error messages. Automatic word wraparound feature is also available.



Lets you move, delete, insert, shift, or replicate individual columns of text. An in-depth review will appear in a future issue of **Ahoy!**
READER SERVICE NO. 133

PaperClip offers a special feature that allows you to manipulate columns of text. This is useful for businesses preparing budgets and financial reports. Columns may be individually moved, deleted, inserted, shifted, and replicated. Files may be saved in the actual screen format or the Pet ASCII format, which allows exchange of information between programs. This format is useful for interfacing your text files with other programs (databases, spreadsheets, etc.), *PaperClip* allows you to use alternate character sets for multilingual applications. Control codes may be sent to the printer for accessing special print functions. Definitely one of the best word processors available for the C-64.

Batteries Included, 3303 Harbor Blvd., Suite C-9, Costa Messa, CA 92626 (phone: 714-979-0920).

QUICK BROWN FOX

Quick Brown Fox; cartridge, $65.00

This menu-driven program requires no printer setup and may be used with a number of 80-column adaptors. The edit function is limited to downward movement and full screen text editing and entering features are not provided. Text is edited one line at a time. Many standard edit functions are available, although their implementation is primitive. DOS functions may be accessed via the clerk menu. Mail merge facility is also available for creating personalized form letters. An advanced file handling feature of the program is the use of boiler-plating, similar to merging contents of a text file at specific locations in another file. The program is very user friendly, and easy to use for the novice. It is not recommended for advanced users.

Quick Brown Fox, 548 Broadway, New York, NY 10012.

SCRIPT 64

Richvale Telecommunications; diskette, $99.95

This word processor features 40/80 column display without the use of any additional hardware. Limited DOS functions are available, so it is recommended that you format your diskettes and configure your printer before you execute this program. Text may be entered one screen at a time. Function keys may be used to access the next screen. Automatic word



Features true word wraparound and audio feedback.
READER SERVICE NO. 134

wraparound and numeric mode functions are also available. Full use of the Commodore 64's cursor control and editing keys is made during the text entering and editing modes.

Text data is saved in the form of screens. You are allowed up to 40 screens per file and 999 screens on a disk. Global search and replace options allow you to search the whole text for a text string and replace it. Printing is done on selected screens in any sequence. The most outstanding feature of *Script 64* is the dictionary provided with it. The 80-column display is poor, unreadable, and not very useful.

Richvale Telecommunications, 10610 Bayview, Richmond Hill, Ontario, Canada, L4C 3N8 (phone: 416-884-4165).

SMITHWRITER

Softsmith Software; diskette, $59.95

This program does not support DOS functions. A separate configurator program adapts the word processor to your particular printer. Text entering is done through the window in the center of the screen which can hold up to 66 characters. An option is available to change this character limit. Full editing features are implemented in this window and lines of text have to be moved into the window for editing. Editing functions such as delete, insert, etc. are available, although poorly implemented. The "push" and "pull" features allow you to move words, phrases, or lines in the file. You can use the rearrange text option to perform block moves. Text may be sent to the disk, tape, or the printer. The unique feature of this program is its ability to print text in more than one column. Good for novice users.

Softsmith, Inc., 2935 Whipple Road, Union City, CA 94587 (phone: 800-341-4000).

TOTL. TEXT®

Totl Software; diskette, $44.00

This is an easy-to-use program written in BASIC (and therefore considerably slower than comparable word processors). It is menu-driven and supports both disk and tape. Text is entered in blocks of 255 characters. Each block is treated as a screen, and function keys are used to move up or down the block to enter and edit text. The most recent version includes the facility to use two disk drives, a feature that increases the processing speed of saving and loading files. Files may be concatenated and printed in a link format. A short machine language program *(Chickspeed)* is provided to speed up the disk access. The print function is well developed and allows you to send special control codes to the printer to access special print functions and commands available with your printer.

TOTL Software, Inc., 1555 Third Ave., Walnut Creek, CA 94596 (phone: 415-943-7877).

WORD PROCESSOR, PROFESSIONAL VERSION

Mirage Concepts; diskette, $99.95

This easy-to-use, menu-driven word processor has a 40/80 column selectable display format. Prior to text entering and editing, the program asks you to specify a few parameters such as lines per page, characters per line, etc. True word wraparound and audio-feedback are the available special features. The edit line is at the center of the screen where all the edit operations occur. In order to perform edit on any line of text, therefore, the line has to be moved into that window. Replace and insert modes are provided to facilitate editing. Block operations may also be performed by manipulating selected blocks of text or data. Blocks may also be saved individually, as compared to the whole file. Print format parameters may be selected from the menu. The 80-column display has poor resolution, and is a disappointing feature. (See full-length review in August *Ahoy!*)

Mirage Concepts, Inc., 2519 W. Shaw #106, Fresno, CA 93711 (phone: 209-227-8369).

WORDPRO3 PLUS/64

Professional Software; diskette, $89.95

This is an excellent, full featured word processor.

It allows up to 352 lines of text in two screens; each one may be selected for the number of lines. This means that you can work on two separate documents, and by using control keys switch back and forth between screens. The alternate screen may also be used to display disk directory or mailing labels for creating form letters. Text may also be transferred between screens. Edit commands are available via the control keys. A command line on top of the screen notes the function or mode currently in use. This line also displays system prompts for additional input from you as well as DOS error messages. Full editing capabilities are available in the text entering and editing mode. Printer configuration is available in the initial start-up options. Special control codes may be sent to the printer for accessing special functions.

Professional Software, Inc., 51 Fremont Street, Needham, MA 02194 (phone: 617-444-5224).

WRITER'S ASSISTANT

Rainbow Computer Corp; diskette, $125.00

This menu-driven program is easy to use and incorporates many standard features. It is, however, slow and not recommended for fast typists. Printer set up and screen colors are accessed from the menu. A disk utility option allows access to DOS functions. Text is entered free form on the screen. Automatic word wraparound feature is also provided. Up to 356 lines of text may be entered before saving it on a disk. Small files may be concatenated on disk to create longer documents. Text may be formatted for printing by using embedded commands. The program offers most standard print formats and functions.

Rainbow Computer Corp., 490 Lancaster Ave., Frazer, PA 19355 (phone: 215-296-3474).

CREATING YOUR OWN GAMES ON THE VIC AND 64

Anyone's Adventures *in* Wonderland

By Orson Scott Card

Any place you can imagine, any adventure you can dream of—you can program an adventure game that brings the dream to life. First of two parts.

I played my first adventure game in a computer store in Salt Lake City. It was a version of the granddaddy of all text adventures, called, appropriately enough, *Adventure*.

In an hour I got lost in a forest, pried open a grid leading down into a cavern, and made my way through the tunnels to a large open chamber with smaller rooms all around, a deep chasm at one end, and a stairway down to another room where a huge snake waited to swallow me whole.

I also died five times, but that's the nice thing about computer adventures. Type RUN and you can live again!

WORDS TELL THE STORY

Computer adventures began with words—and nothing but words. You don't actually see the cave you're exploring on the screen in front of you. Instead, you read descriptions of each room you enter. Also, instead of a joystick, you move around with more words.

Let's play a sample game. The words the player types into the computer will be in *ITALICS;* the computer's words will LOOK THIS THIS. This game is called *Sky Scraper,* but don't look for it in your nearest computer store. I'm making it up as I go along.

The packaging of the game says that you're a nearly bankrupt private investigator who just got a $5,000 retainer to find out some information about GAK, a major computer firm, which is doing some vital research in the Wellman building, a skyscraper in Manhattan. At the beginning of the game, you are in the foyer of the Wellman building.

FOYER
YOU'RE STANDING IN THE CROWDED FOYER OF AN OFFICE BUILDING. PEOPLE ARE WALKING QUICKLY ALL AROUND YOU, HEADING INTO A BANK TO THE WEST, A SMALL SHOPPING MALL TO THE EAST, OR THE ELEVATOR LOBBY TO THE NORTH. THERE'S AN INFORMATION BOOK RIGHT BY YOU, BUT THERE'S NOBODY THERE.
LOOK BOOTH
THERE ARE BROCHURES IN A RACK ON THE COUNTER AND A SMALL BELL TO SUMMON HELP. IN A HALF-OPEN DRAWER BEHIND THE COUNTER YOU CAN SEE SOMETHING SHINING.
RING BELL
DING.

(That was exciting. Well, I'm an investigator—better start snooping.)

OPEN DRAWER.
THE DRAWER IS NOW OPEN
LOOK DRAWER
IN THE DRAWER YOU CAN SEE A LETTER OPENER, A RING OF KEYS, A FLASHLIGHT, AND A PAPER BAG THAT LOOKS LIKE SOMEBODY'S LUNCH.

(These sound pretty useful—better take them along.)

TAKE ALL

TAKEN
INVENTORY
YOU HAVE THE FOLLOWING ITEMS:
LETTER OPENER
KEYS
FLASHLIGHT
LUNCHBAG
BROCHURE
BELL

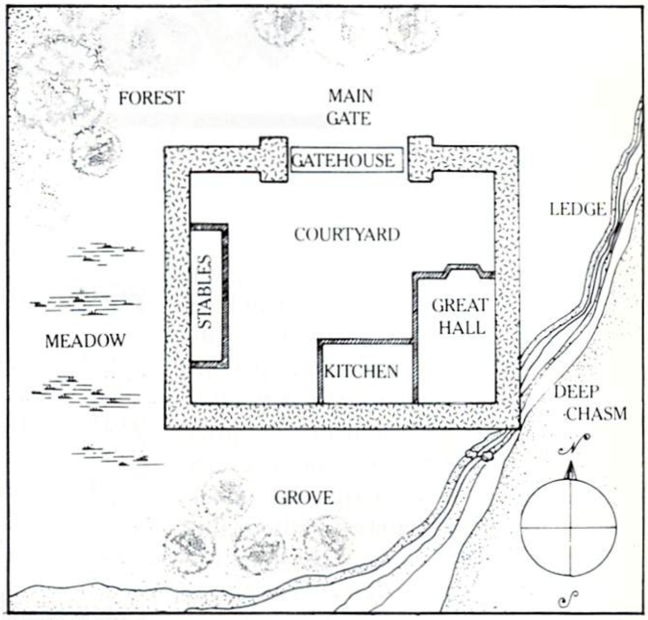
(I got more than I bargained for—I meant to take only the items in the drawer. But as long as I've got a brochure, I might as well see what it is.)

READ BROCHURE
"WELCOME TO THE WELLMAN BUILDING. THIS HISTORIC STRUCTURE WAS BEGUN IN 1929, AND THE BASIC CONSTRUCTION WAS FINISHED BY MID-1930. HOWEVER, THE GREAT DEPRESSION DRIED UP THE FUNDS FOR COMPLETION, AND THE BUILDING STOOD UNOCCUPIED, EXCEPT FOR A FEW TENANTS ON THE LOWER FLOORS, FOR 11 YEARS, WHEN AMERICA ENTERED WORLD WAR II. AT THAT TIME THE NAVY USED SEVERAL FLOORS FOR WEAPONS RESEARCH AND DEVELOPMENT, WHILE ANOTHER FLOOR WAS USED FOR THE CITY'S INDUSTRIAL MOBILIZATION EFFORT. AFTER THE WAR, THESE FLOORS CONTINUED TO BE LEASED BY THE GOVERNMENT, THOUGH THEY WERE LOCKED UP AND NEVER USED. MOST OF THE BUILDING WAS NEVER OCCUPIED, AND ITS OWNER MADE NO EFFORT TO LEASE THE SPACE.
RECENTLY, THE OWNER CONTRACTED WITH F.G. IVUUPO TO RENOVATE AND LEASE MOST OF THE PROPERTY. THE RESULT IS THE ATTRACTIVE WELLMAN MALL, WITH DOZENS OF INTERESTING SHOPS AND RESTAURANTS; THE WELLMAN BRANCH OF CITYWIDE BANK; AND 37 FLOORS OF OFFICES TO OPEN TO THE GENERAL PUBLIC.

(On the off chance that there's more information, I'll read again.)

READ BROCHURE
THE WELLMAN BUILDING IS A FINE OLD SKYSCRAPER ON THE OUTSIDE, A PLUSH MODERN OFFICE BUILDING ON THE INSIDE. THE BUILDING IS 644 FEET FROM STREET LEVEL TO ROOFTOP. EACH FLOOR WAS ORIGINALLY 14 FEET FROM THE FLOORS ABOVE AND BELOW, BUT FOR ENERGY EFFICIENCY ALL THE CEILINGS HAVE BEEN DROPPED TO A MORE STANDARD, BUT STILL SPACIOUS, 9 FEET ABOVE THE FLOOR, ALLOWING THE INSTALLATION OF STATE-OF-THE-ART SOLAR HEATING AND COOLING EQUIPMENT.
WE HOPE YOU WILL ENJOY YOUR VISIT TO THE WELLMAN BUILDING. IF YOU HAVE ANY QUESTIONS, PLEASE ASK ONE OF OUR COURTEOUS GUIDES IN THE INFORMATION BOOTH.

(Ah. There *was* more. And all these figures might add up to something. Let's see, each floor is 14 feet high; the building is 644 feet high; that means there could be as many as 46 floors. Even allowing for a foyer twice the normal height, that still means that with "37 floors of offices open to the general public," there might be a few floors *not* open to the general public. Also, there is a crawl space between the original ceiling and the new lowered ceiling on every floor—that might come in handy sometime.)



FLOOR PLAN

Layout of the Castle of Darkness (see later pages).

(Now, do we try the bank, the shops, or the elevators?)

GO WEST
THERE IS A MAN LOCKING THE BANK DOOR. HE FROWNS AT YOU AND SAYS, "THE BANK IS CLOSING."

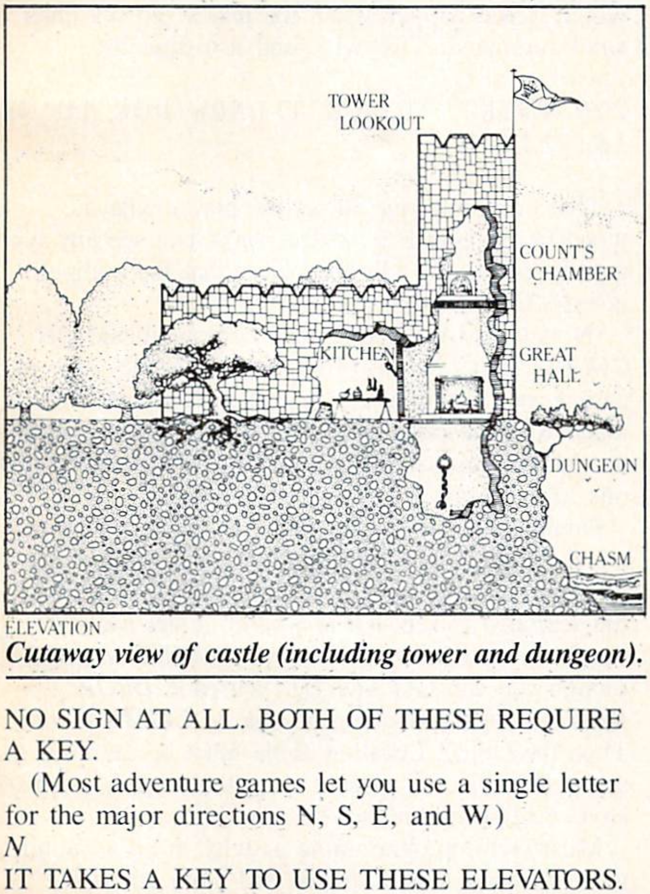
(Sometimes in adventure games it pays to try more than once.)

GO WEST
THE MAN SAYS, "COME BACK IN THE MORNING."

(Well, persistence *does* make a difference. Let's try it again.)

GO WEST
THE MAN BECKONS TO THE SECURITY GUARD, WHO STARTS TOWARD YOU, HIS HAND ON HIS GUN.

(This doesn't look good. Let's go somewhere else.)
GO NORTH
ELEVATOR LOBBY
THE ELEVATORS ON THE WEST SIDE GO TO FLOORS 2-21; THOSE ON THE EAST SIDE ARE EXPRESS TO FLOORS 21-40. AT THE NORTH END OF THE LOBBY ARE TWO MORE ELEVATORS: A SERVICE ELEVATOR AND ONE WITH



Cutaway view of castle (including tower and dungeon).

NO SIGN AT ALL. BOTH OF THESE REQUIRE A KEY.

(Most adventure games let you use a single letter for the major directions N, S, E, and W.)
N
IT TAKES A KEY TO USE THESE ELEVATORS. WHICH KEY WOULD YOU LIKE TO TRY?

(How do I know which key? Forget it for now.)
E
THE ELEVATOR HAS BUTTONS FOR THE FOLLOWING FLOORS: P1 P2 P3 P4 B L 20 21 22 23 25 26 27 28 29 30 31 33 34 35 36 37. WHICH BUTTON WILL YOU PUSH?

(Aha! Notice that there are four parking levels and a basement. Also notice that there is no button for 24 and no button for 32. But the game doesn't point this out—you're a private investigator. You have to notice these details for yourself. They may be important; they may not.)
24
THERE IS NO BUTTON 24.

(OK, I was trying to be cute. I'll push a real button now.)
37
THE ELEVATOR SOARS UPWARD AT AN ALARMING SPEED.
THE DOORS OPEN INTO THE RECEPTION AREA OF "GAK COMMERCIAL ACCOUNTS".

There are many possibilities from here on. Many floors are probably ringers—nothing can happen there except that you get in trouble or waste time. But the service elevator, the missing floors, the unidentified elevator, the crawl spaces, the shops in the mall, the bank, the parking levels—there's a lot of space to be explored in this building. The encounter with the man at the door of the bank suggests that there are also risks. There might be unfinished floors; you might find the floors that have been sealed off since the Navy research in World War II; you might find some very odd secrets indeed.

Other adventure games provide other opportunities. Treasure islands, underground empires, space stations, murder mysteries, haunted houses—if you can imagine a place, you can put an adventure in it.

WHAT ARE SOME GOOD ADVENTURE GAMES?

Perhaps the most popular of all adventure games is Infocom's three-part (so far) *ZORK* series. I have played only *ZORK I* to exhaustion, but I can assure you that its popularity is well deserved. Infocom specializes in text-only adventures—games that use nothing but words—and they are all excellent, partly because of the best parsing routines in any adventure games.

That deserves a bit of explanation, because you're going to face the problem of parsing in the adventure games you program. Parsing is the process of taking the instructions the player types into the computer and interpreting them. When the instruction is as simple as *N* for *north,* it's easy enough. Most adventure games allow you to use only two-word commands, beginning with a verb and ending with a noun or adverb:

GO NORTH
TAKE KNIFE
DROP LAMP
KILL DWARF
OPEN DOOR
LIGHT LAMP

Parsing these simple commands is hard enough. The program has to cut the command string into the

two parts, then decide whether it is a legal word and act on it. This is done by searching a table.

TABLES AND SEARCHES

A table is a list, in a deliberate order, stored in memory. You conduct a search by examining all the legal command words in the list until you find a match with the word the player typed, or find that none of the words match.

The command word table would include all the legal verbs and all the legal one-word commands: N S E W NE SE NW SW GO TAKE LOOK DROP KILL OPEN LIGHT. . .CLIMB. In a BASIC program, you would put each of these command words in a string array, so that CW$(0) would be "N", CW$(9) would be TAKE, and CW$(22) might be CLIMB.

But it isn't enough to list the words in the table. They have to be listed in a useful order. That's because each table is related to other tables, which will be in a related order. So the table gives us two items of information: not only the name of the command, but also its position in the table. "N" is therefore command 0; "TAKE" is command 9; and "CLIMB" is command 22, because each comes in that order in the table.

Let's say the player has typed *CLIMB WALL*. That was INPUT into the string PI$. Then it was cut into two words, P1$ and P2$. Now we want to see if P1$, the first word, matches any of the commands in the table CW$*(n)*. This subroutine would carry out the search:

```
500 CM=23:FOR I=0 TO 22:IF P1$=CW
$(I) THEN CM=I:I=23
510 NEXT:RETURN
```

The routine is simple. CM, the command code, is set to 23. Then P1$ is compared to each entry in the CW$ array. If there is a match, CM is set to the number of the command (its order in the table, remember), and *I* is set to 23 so the loop will end immediately. If no match is found, then when the loop ends, CM will still have the value of 23.

JUMP TABLES

Now that we know which command the player has called for, we go to another table—a jump table. Each command is carried out by a particular subroutine in the program. We get to the right subroutine by using a simple jump table, like the one in this statement:

```
ON CM+1 GOSUB 250,550,1000,1100,1
200,600,650,670,900...2500,200
```

The value of CM, which represents the *position* of the command word in the command table, is also the position in the jump table of the line number of the routine that executes that command. Since CLIMB was command word 22, then it will be executed when the program jumps to the 22nd line number folowing the GOSUB statement.

Notice that even though there are 22 commands, there must be 23 line numbers after the GOSUB. That's because CM might have the value of 23, which would happen when the player did not enter a valid command. That will send it to line 200:

```
200 PRINT "I DON'T KNOW HOW TO "P
1$:RETURN
```

That error message shows the player what he typed by printing PI$, so the player can see any typing errors. It also tells the player that the computer is expecting a verb.

(Notice also that the sample command said ON CM+1. That's because CM has a possible value of zero in our example, and an ON *n* GOSUB command won't respond to a zero. The solution is either to make it impossible for a command to be 0 or add one to the number.)

Parsing has only begun, however, when the command has been recognized. Then the second word, if there is one, has to be analyzed, and an Object Table searched to see if it is a valid object name *for that command* (you obviously can't *TAKE NORTH,* though you can *GO NORTH;* you can't *DROP WALL,* though you might be able to *CLIMB WALL*). Then the Object Location Table must be searched to see if the object is present or in the player's possession. And so on and so on.

Most of the programming activity in an adventure game involves table searches and table jumps.

And that's why Infocom's parser is so remarkable. Instead of two-word commands, it is quite possible in an Infocom game to type a command like this:

```
KILL THE TROLL WITH THE SWORD,
TAKE THE AX, GO EAST AND REST
```

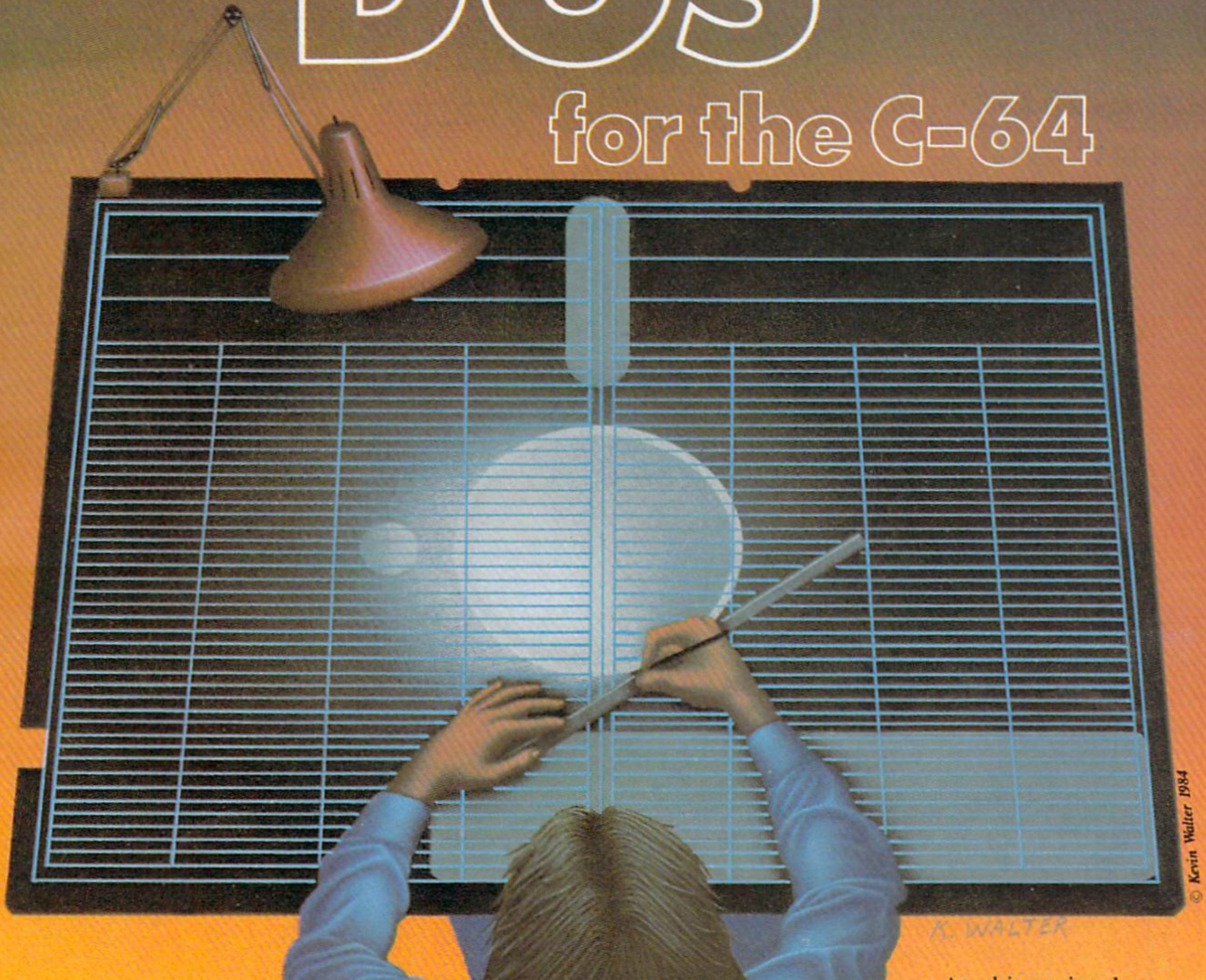
This is starting to sound like English, isn't it?

Yet, in my opinion, the parser, good as it is, isn't the main value of the Infocom games. The best thing about them is that they are good *stories.*

That's one of the curious things about adventures —they have a lot in common with games, it's true,

Continued on page 42

DOS for the C-64



By Bob Lloret

For all you beginning programmers who can't quite get a handle on all those Disk Operating System commands and all you more experienced programmers who, like myself, don't like the wedge, I've created *DOS*. *DOS* incorporates all the normal disk commands used by a programmer and reduces them to one keystroke from a main menu, making your life that much easier. It also includes a HELP screen and a disk directory which will display and run any program on that disk with one stroke of a finger.

After typing in the program, save a copy to disk or tape in case of any typing error, then type "RUN" and you'll get a brief look at the title screen. Once this is done, the main menu will appear and you will have a choice of seven options: FORMAT DISK, RENAME FILE, VALIDATE DISK, ERASE PRG., WRITE DOS FILES, DIRECTORY and HELP SCREEN.

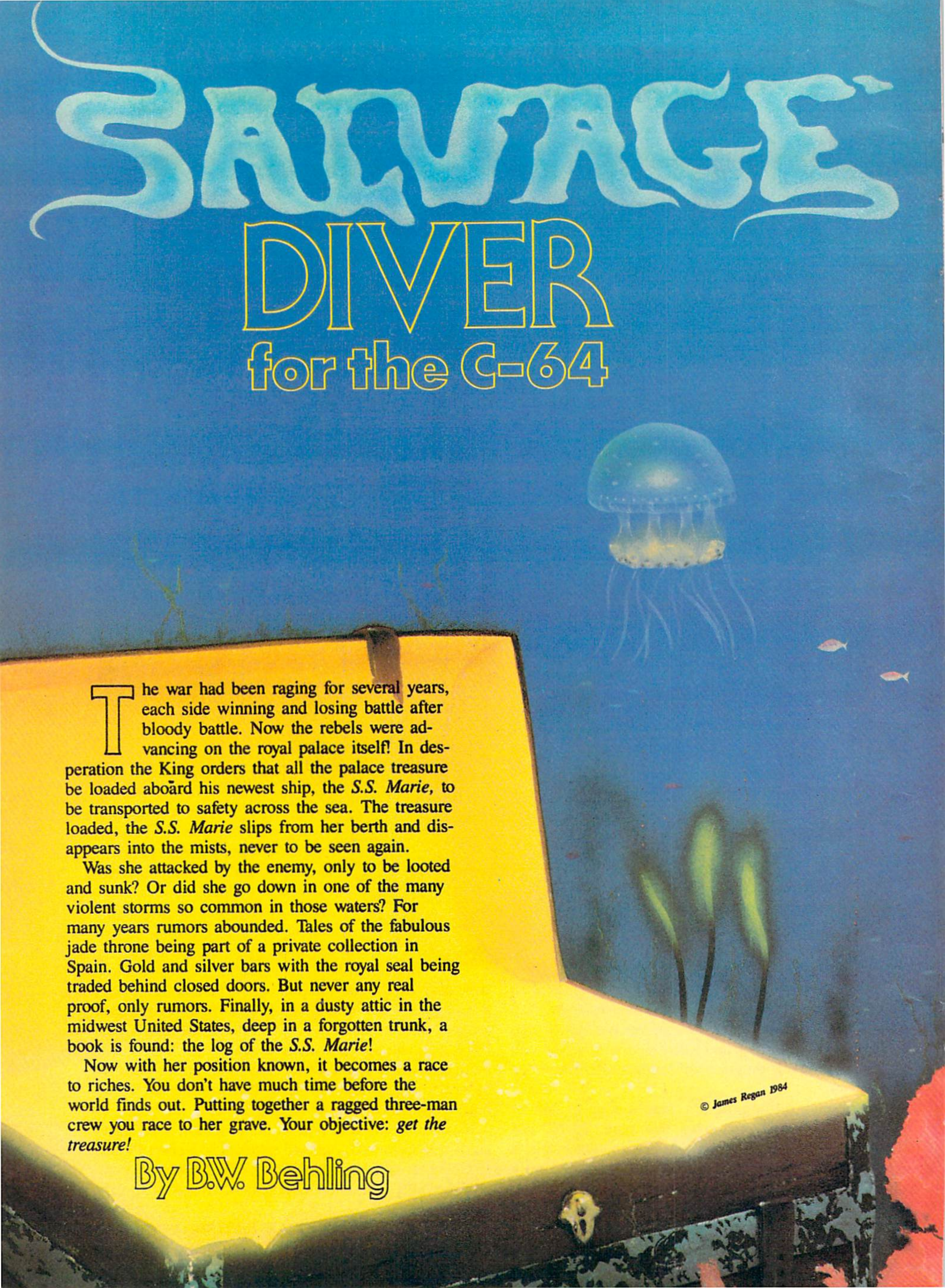
At this point let me strongly suggest you go straight to the HELP SCREEN. This will explain all your other options. Prompts have been used when necessary in order to make the program easier for the beginner to use.

After I completed the program, a friend of mine just getting started with his C-64 came to me asking for help with the disk commands. I gave him a copy of *DOS*. The next day he couldn't thank me enough and praised the program. It felt great to know I helped another person, and I hope I can do the same for you. Put a copy on all your disks with option #5 so it will be handy when you need it. One more point: because of its name, you will not see it listed on the directory, but be assured it's there waiting to work for you.

If you don't wish to type in the program, send $5.00 and a self-addressed, stamped envelope to Robert Lloret, 157 Atlantic Avenue, Staten Island, NY 10304 and a copy will promptly be mailed to you. Don't forget to specify disk or cassette!

SEE PROGRAM LISTING ON PAGE 82

SALVAGE DIVER
for the C-64



The war had been raging for several years, each side winning and losing battle after bloody battle. Now the rebels were advancing on the royal palace itself! In desperation the King orders that all the palace treasure be loaded aboard his newest ship, the *S.S. Marie,* to be transported to safety across the sea. The treasure loaded, the *S.S. Marie* slips from her berth and disappears into the mists, never to be seen again.

Was she attacked by the enemy, only to be looted and sunk? Or did she go down in one of the many violent storms so common in those waters? For many years rumors abounded. Tales of the fabulous jade throne being part of a private collection in Spain. Gold and silver bars with the royal seal being traded behind closed doors. But never any real proof, only rumors. Finally, in a dusty attic in the midwest United States, deep in a forgotten trunk, a book is found: the log of the *S.S. Marie*!

Now with her position known, it becomes a race to riches. You don't have much time before the world finds out. Putting together a ragged three-man crew you race to her grave. Your objective: *get the treasure!*

By B.W. Behling



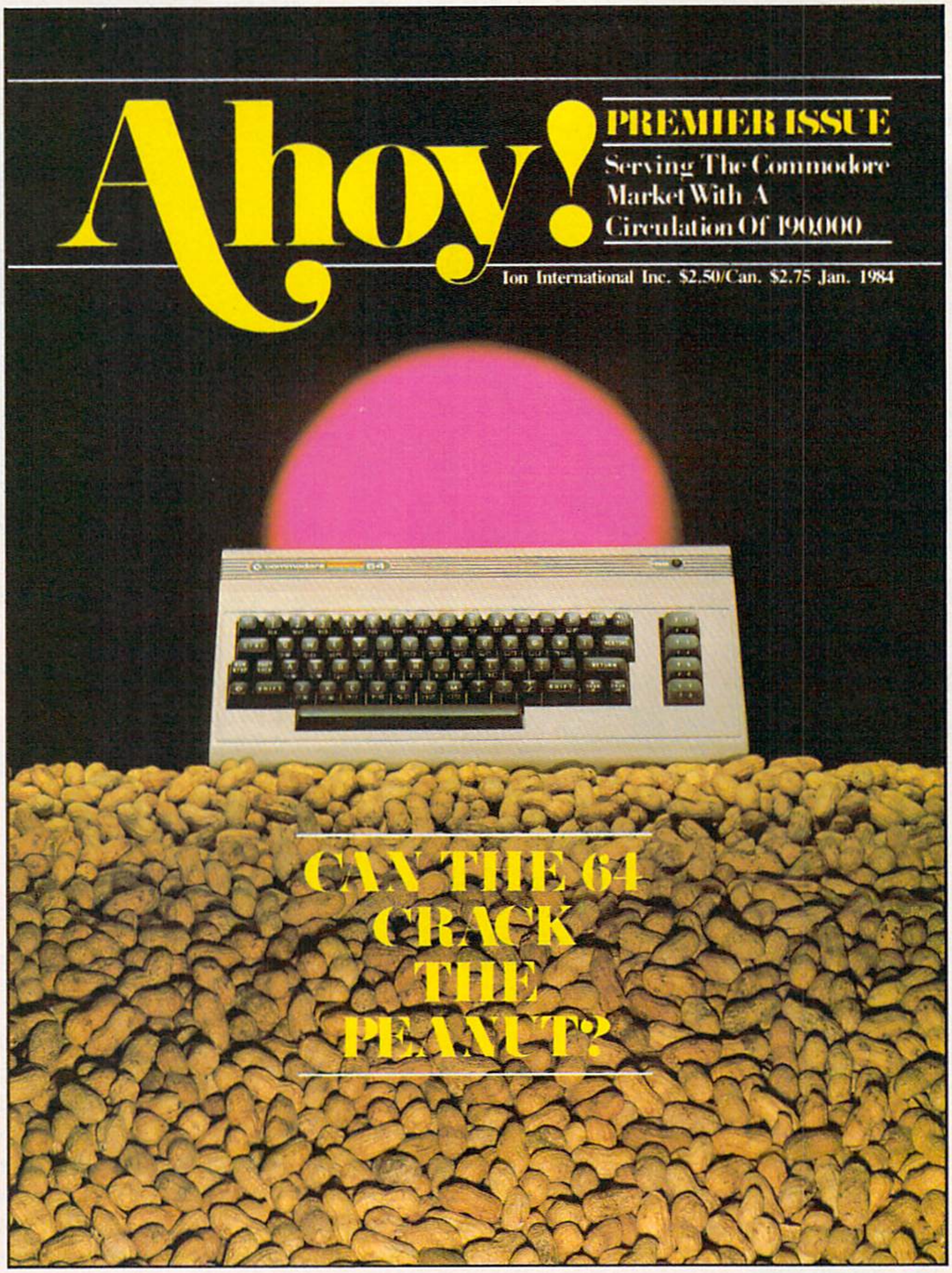
(Note: Any program can bomb if lines are entered incorrectly. Those programs using machine language subroutines are especially vulnerable to crashes. Since many Commodore 64's automatically run programs when they are loaded, I have included a stop command in line #1. This way, if the program crashes when you run it, you can reload it without it automatically crashing again. To run the program after loading, simply type RUN2 or type RUN and then CONT after the break in 1 message.)

When the game begins you start with 3 divers, each with a full tank of air. Controlling the diver with a joystick in Port 2, you must dive to the ocean floor and find the X-marked treasures, all the time avoiding the hungry sharks. When your diver grabs a treasure, it will be immediately identified and its value will be displayed and added to the diver's total. The longer the diver stays down the lower his air supply gets and the faster the sharks move. When the air gauge turns yellow your air is getting low. When the gauge is red you are on your reserve supply. If your air gets below 100 lbs., the diver will move slower due to low oxygen. If you run out of air the diver will drown and all his treasure will be lost. If the diver completes his dive and returns to the surface the value of his treasure is added to the net total and a new diver swims out to take his place. Any time a shark contacts a treasure he will swallow it and a new treasure will be marked.

If you would like a copy of this or any of my programs on *cassette only,* send a blank tape and a self addressed stamped mailer along with $5.00 and the program name to:

B.W. Behling
232 Jackson Street
Brooklyn, NY 11211

SEE PROGRAM LISTING ON PAGE 84


Ahoy!
PREMIER ISSUE
Serving The Commodore Market With A Circulation Of 190,000
Ion International Inc. $2.50/Can. $2.75 Jan. 1984
CAN THE 64 CRACK THE PEANUT?

I don't really hate to write letters, but it seems like I'm always behind in my correspondence. For example, at least half the Christmas cards that should be sent to my far-flung relatives never are. But thanks to my Commodore 64 I not only sent out all my cards last Christmas, I also enclosed a personalized letter in each card. My aunt in Florida loved it. My mother loved it too, because for once she didn't have to make excuses for her tardy son.

Letter writing like this is an obvious use for a word processor. Run a copy of the letter on the printer, change the greeting, run another copy and so on. A 'mail merge' program will even address the envelopes.

However, the more you use a word processor the more not-so-obvious applications you discover for it—some trivial, some not so trivial. Here, culled from my personal experience, are

TEN USES FOR YOUR WORD PROCESSOR

BY ED HOORNAERT

1.

Make your own stationery.

People like to have their possessions personalized—witness automobile license plates and T-shirts with names on them. Personalized stationery is useful, too, but expensive. Besides, you usually have to buy about 500 sheets at a time!

Computers, however, are ideal for this sort of task. Set up the letterhead you want on the screen. If your printer does fancy printing—bold face, italics, subscripts, double size letters or the like—experiment till you get a distinctive result. Then use the word processor's editing capabilities to make sure it's perfect. Use your imagination!

Most word processing programs let you print more than one copy. Set it to 25 or 50 copies, feed in individual sheets of quality paper and you're all ready to catch up on your correspondence.

2.

Keep your address book up to date.

People move around a lot these days. My address book always used to be filled with addresses scratched out and scribbled in. Worse yet, I sometimes had no idea which of two or three addresses was the right one.

A computer makes updating such records a snap. The first time around you'll have a bit of typing, but if you save the list on a disk or tape you'll never have to do the whole thing again. When you have a sufficient number of new friends or new addresses, load your file. If your word processor has a 'search' function, use it to find the name of your peripatetic friends. Simply type the new address and phone number over the old, then sit back and watch your printer do the rest of the work.

3. Keep track of collections.

Do you collect records? Baseball cards? Maybe plates are your thing. A word processor can help you organize and keep track of any kind of collection, from antiques to zippers.

For example, you might start a word processing file called Classical Records. You could label the first section "Beethoven" and then list all his music—along with orchestras, conductors, and record labels if you wish. Then do the same with Brahms, Britten, and so forth. To add a symphony by Gustav Mahler, use the search function to find "Mahler" and insert the information in the proper place. In effect, you can use your word processor as a mini-database!

4. Keep insurance files.

Have you memorized the serial number of your computer? I haven't. I'm not even sure where to find it. (I just checked. It's on the bottom of my Commodore—P00221493.) Yet if your home is robbed or burned, the police and insurance agency might want that number, as well as a detailed list of other valuables in each room.

Once again, your word processor makes the chore easier. The hardest part is figuring out all that you have and collecting serial numbers. When you're done typing your list make sure that you save it on a tape or disk, because you'll need to revise it periodically. But don't worry about that. Changing or adding to a list is easy.

One last suggestion about your insurance list. Keep it in a safe place, but not in your home. The ashes of the list wouldn't help you a bit.

5. Write more effective resumes.

Once upon a time, many people stayed at the same job their whole lives. Not anymore. And if you're the one changing jobs the computer can help you write a more effective resume.

A resume is like an ad. The product advertised is *you*—your skills and experience. With a word processor, it's easy to change the basic format of a resume, stored on a disk, to highlight your suitability for a particular job. A tailor-made resume, preferably run out on a letter-quality printer, presents you in the best possible light for each job opening.

Imagine, for example, that you want to be a staff programmer for a software publisher. One company produces game software for young children, so you would load your "basic resume" from disk and add specifics about your experience with youngsters. A second company, however, produces business software. The resume sent to them would discuss in detail your business background.

Maybe you'll owe your next job to your computer.

6. Make your opinions count.

Lots of people say they don't care about politics. But they still have opinions about political issues—maybe disarmament, or the construction of a new factory near home. If you're like me, though, you rarely get around to writing the letter to a newspaper or elected representative that might help your cause.

A word processor, however, can be even more effective than your vote in influencing public policy. A letter can be personalized and sent to several newspapers and public officials. With almost no extra effort, your opinions spread farther—and carry more weight.

7. Stand up to the big guys.

Sometimes a letter to a politician isn't enough. And at these times your word processor will help even the odds between you and the 'big guys.'

Richard Bach, author of the bestselling *Jonathan Livingston Seagull,* moved to a quiet valley in Oregon only to find that logging companies were going to denude the hillsides. Bach and a group of neighbors researched legal objections to the logging. Using database programs and a word processor, they produced a 600-page protest that forced the withdrawal of the timber sale.

Furthermore, the protest was easily adapted to fight other controversial timber sales, putting the private citizens on a more equal footing with big

business and government agencies. The bulk of the work needed to be done just once, thanks to word processing.

8.

8. Write a painless newsletter.

It seems like every group sends out a newsletter—schools, computer users groups, clubs, businesses—and usually a reluctant volunteer is saddled with the chore. A word processor makes a professional-looking newsletter as easy as typing 200 words a minute (on your printer, of course).

The computer lets you catch and correct mistakes on the screen, rather than on paper. Even hunt and peck typists can produce a perfect copy. So why not volunteer? Who knows, you may be tempted to produce a newsletter just to show off your word processor!

9.

9. Help your kids get ahead in school.

My eight year old son has a homework assignment to write about the ways computers can be used. He wants to use the word processor to get a finished copy that looks far better than he can do by hand. Even kids want their work to look great.

Furthermore, your children will be learning valuable lessons in computer applications. Word processing will be an important skill in their lifetime.

And once they learn the keyboard and how to edit, an added bonus may come about. The computer removes the physical blocks to making changes and improvements. If they wish to add a sentence in the middle of an essay, for example, they don't have to recopy the whole thing. Thus they may become more willing to go back and improve their work. Word processing can be a boon even to young computer users.

10.

10. Unleash your creativity.

What works for kids works for adults, too. As you become comfortable with word processing you, too, will be more willing to change and improve your writing. Furthermore, the printer does such a neat, professional job that you'll marvel that you really wrote the piece.

So, do some writing of your own—maybe a short story, or a diary, or a poem, or a magazine article. What the heck, why not try a book? See if your word processor's easy correctability can break down your inhibitions, born of laborious recopying in school. Be creative!

It's true. Your Commodore can help you be creative. And an example of that creativity just might be the new uses that *you* develop for that wonderful tool of the computer age—your word processor.

Morton Kevelson's analyses of the 1541 and MSD SD-2 disk drives have become industry standards. And in the October and November issues of *Ahoy!* he provides the same expansive coverage of an even more expansive subject: graphics on the Commodore 64. Included will be articles on bit map and character graphics, reviews of available programs, and original graphics routines.


© James Regan 1984

Unraveling the Mysteries of Sound on the C-64

Part 2 Sound Explorer

by David Barron

Last month I spoke of the various parameters that control electronically synthesized sound on the Commodore 64. This month I present *Sound Explorer,* a program that will let you play around with the three oscillators in the 64.

Each oscillator has the capability of producing one of four waveforms:

1) Noise: a random waveform that can be used for gunshots or crowds cheering.

2) Pulse: Basically a square wave with an adjustable "duty cycle." The duty cycle refers to the width of the squares. The width is adjustable by "Pulse width." (This control will only work when the waveform is set on "Pulse.")

3) Sawtooth: this waveform produces a very raspy sound that may prove quite useful.

4) Triangle: a very mellow waveform.

Full control of the envelope is provided via ADSR parameters. In all cases, 15 represents the slowest value and 0 the fastest (attack, decay, release). To control sustain, 15 is the loudest value and 0 is the lowest.

The "ON=1" parameter determines whether the oscillator is turned on or not. A "1" means it will be on and a "0" means that it is off.

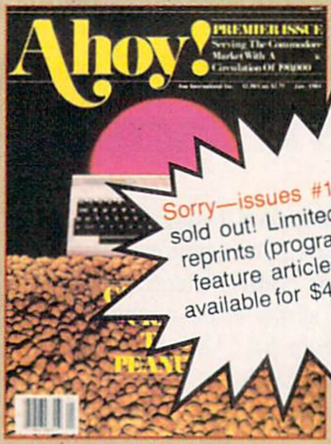
Bandpass, lowpass, and highpass filters are provided. The cutoff frequencies of these filters may be varied by using the "Cutoff Low" and "Cutoff High" parameters.

The general operation of the program is quite simple. Once the program is loaded and run, all the

Continued on page 75

Back Issues of Ahoy!

Sorry—issues #1 and #2 sold out! Limited edition reprints (programs and feature articles only!) available for $4.00 each.



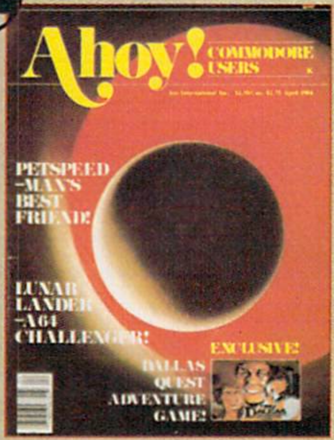
ISSUE #1—JAN. '84 $4.00
The 64 v. the Peanut! The computer as communications device! Protecto's Bill Badger interviewed! And ready to enter: the Multi Draw 64 graphics system! The Interrupt Music Maker/ Editor! A Peek at Memory! Programming Sequential Files!



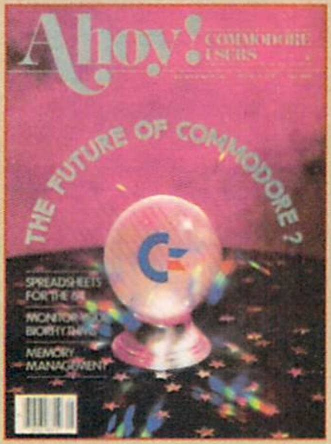
ISSUE #2—FEB. '84 $4.00
Illustrated tour of the 1541 disk drive! Synapse's Ihor Wolosenko interviewed! Users groups! Artificial intelligence! And ready to enter: Music Maker Part II! Night Attack! Programming Relative Files! Screen Manipulation on the Commodore 64!



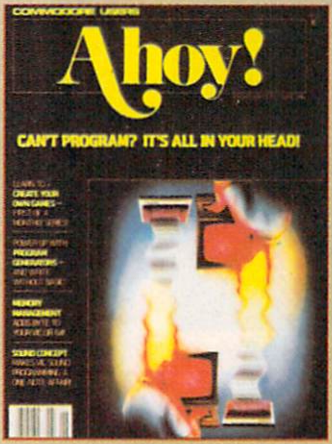
ISSUE #3—Mar. '84 $4.00
Anatomy of the 64! Printer Interfacing for the 64 and VIC! Educational software: first of a series! Commodares! And ready to enter: Space Lanes! Random Files on the 64! Easy Access Address Book! Dynamic Power for your 64!



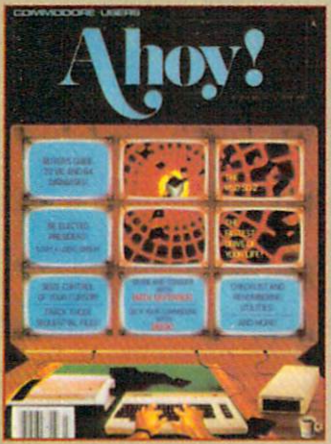
ISSUE #4—APR. '84 $4.00
Petspeed and Easy Script tutorials! Printer interfacing and educational software guide continued! Lower case descenders on your 1525! Laserdisc! The Dallas Quest Adventure Game! And ready to enter: Apple Pie! Lunar Lander! Name that Star!



ISSUE #5—MAY '84 $4.00
The Future of Commodore! Inside BASIC program storage! C-64 Spreadsheets! Memory Management of the VIC and 64! Educational Software Guide continues! And ready to enter: Math Master! Air Assault! Biorhythms! VIC 20 Calculator!



ISSUE #6—JUNE '84 $4.00
Game programming column begins! Program generators! Rupert on inputting! Memory Management and Educational Software Guide continue! And ready to enter: Post Time for the 64 and 20! Alpiner! Sound Concept for the VIC 20!



ISSUE #7—JULY '84 $4.00
The MSD Dual Disk Drive! Database buyer's guide! The File Sleuth! Creating your own games! Training your cursor! Users groups! Commodares! And ready to enter: Checklist! Renumbering! What's My Job? Brisk! Math Defender! More!



ISSUE #8—AUG. '84 $4.00
Choosing a word processor! Unraveling the mysteries of sound on the 64! Computational wizardry! Creating your own word games! And ready to enter: Micro-Minder! Directory Assistance! The Terrible Twins! Words Worth!

Send coupon or facsimile to:
Ahoy! Back Issues, Ion International Inc., 45 West 34th Street—Suite 407, New York, NY 10001

Ahoy!

Please Send Me The Following:
____ Copies of issue number ____
____ Copies of issue number ____
____ Copies of issue number ____

Enclosed Please Find My Check or Money Order for $________

(Outside the USA please add $1.00 for every copy)

NAME ____________
ADDRESS ____________
____
CITY ____________
STATE ____________
ZIP CODE ____________

Building a Cassette Interface

For Your Commodore 64

The Commodore 64 came into our home like a naked baby. It had neither disk drive nor Datasette. It played its little games on the TV screen, but its miniature programs that I wrote for it were lost forever on power dówn. I already had three disk drives and a cassette recorder hooked up to my Radio Shack TRS-80 Model 1 computer, so why did I need more? My vision was to see the Commodore and the TRS-80 talking to each other. So my first priority was to be able to save programs. I thus built a small and inexpensive cassette interface that I could use with my own recorder. This article shows how I did it.

By Anthony T. Scarpelli



© Denise Satter 1984

I have been designing electronic circuitry for a long time. It's my vocation and avocation. I couldn't see spending $60 for another cassette recorder, especially since I knew that I could build an interface for less than $15. In fact, I built it for $13, and most of the parts came from our local Radio Shack stores, whose parts are not always inexpensive.

If you are the least bit electronically inclined, or know of someone who is, you can save yourself a number of dollars if you build this device, especially if you already have a cassette recorder. Your recorder doesn't have to be an expensive unit either. The method used to store and retrieve the data is fairly immune to noise, volume levels, and saving and loading problems, and should be able to work with any unit. I say 'should' since I haven't tried it with any recorder other than my own Realistic CTR-35. But the tapes made with the interface were able to be used without any problem on the Datasette, and vice versa.

The specifications for the interface required that it work exactly like the Datasette. This meant that it be comparable with the Datasette and its tapes, and that it start and stop under computer control. Also, it should be reliable. Back in the old days of the Model I (1977), when the cassette recorder was the only way to save programs, the tapes were notoriously prone to poor loads. The loading of a program from tape was at times a test in patience. The volume level was very critical, and if it wasn't just right, you could spend hours trying to retrieve a program made on another machine.

The first job I had was to learn how the Commodore connected to the outside world. One of the many ports that are available is the cassette port. Its pinout is shown in Appendix I in both the *Commodore 64 Programmer's Reference Guide* and the *Commodore 64 User's Guide.* With this pinout, the schematic of the C-64, some standard test equipment, and a friend who had a Datasette, I discovered the deep, dark secrets of Commodore cassette interfacing.

The pin functions of the port work like this:

Pin A-1 is ground. Ground is always the reference connection in any piece of electronic equipment. All signals and power voltages are measured from it.

Pin B-2 is plus five volts (+5v). This is the same voltage that actually powers the Datasette as well as the computer.

Pin C-3 is the cassette motor connection. This pin is connected to the emitter of a transistor in the computer, and is connected to the motor in the Datasette. A signal from the computer turns the transistor on and off, just like a switch. Thus the motor will turn on and off.

Pin D-4 is called the Cassette Read pin. This pin is an input to the computer. The computer 'reads' the signal that comes from the recorder. The signal is an amplified voltage of the minute signal that had

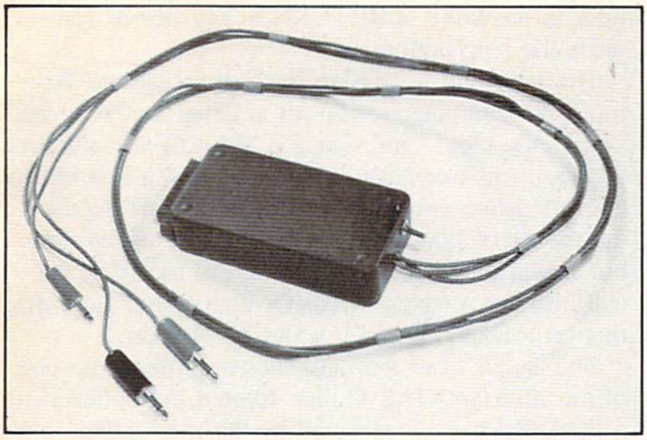


Photo 1: outside of the assembled cassette interface.

been stored on the tape. It is, and always should be, at a five volt amplitude.

Pin E-5 is called Cassette Write. This signal comes from the computer and goes to the recorder. The signal represents all the bits and bytes that you want to save in your program. It is also at a five volt amplitude and stays at that level all through the Datasette. It actually drives the recording head in the unit after some inversions.

Pin F-6 is the Cassette Sense input. It is the line that tells the computer whether the Datasette is on or off. A switch in the Datasette will bring this line to the ground level whenever the play switch is pressed. If the computer doesn't see a low level here, it will display the 'Press Play. . .' message whenever you load or save a program.

Now that we know what all the pins do, the next step in designing the interface is to work with each line and get some circuitry built that does what the Datasette does. The easiest line to start with is the ground connection. There is a straight connection from the computer to the interface to the cassette recorder. Nothing else need be done.

The next line is the +5v line. This voltage will power all the circuitry in the interface, but will not go to the recorder.

The motor pin is driven from the computer's nine volt power source, and is somewhat regulated to 7.5 volts. Two problems are associated with this line. One is that the 7.5 volts is not always 7.5 volts,

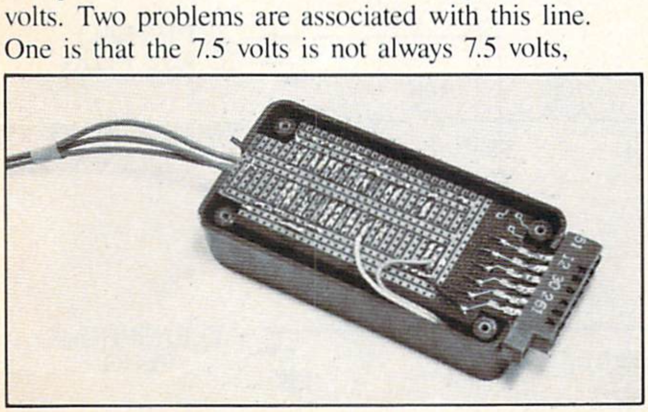


Photo 2: Inside interface, showing bottom of board.

especially when you put a load on it. In fact, I measured only 6.5 volts on this line. This is a nice voltage for motors in many cassette recorders, but not all of them. So the second problem is how to make this line independent of the cassette recorder's own voltage. This is simply done by the use of a small reed relay. Refer to the schematic to see how this is done.

The 6.5 volts comes in through the C-3 pin, is dropped slightly by a resistor, and drives the 5 volt relay. The diode connected across the relay protects the transistor in the computer from being destroyed by a high voltage spike that is produced by the coil of the relay.

The contacts in the relay are connected to the 'remote' line of the cassette recorder. There is also a switch across this line to allow the recorder to be turned on for fast-forwarding or rewinding. Normally the computer controls the recorder. But when the computer has turned the recorder off, the only other

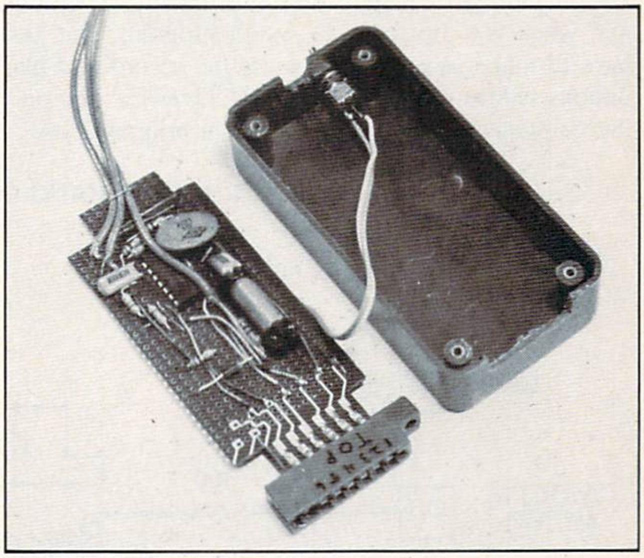


Photo 3: Layout of the components in the interface.

way to control the recorder yourself would be to unplug the remote plug. This is boring after a while, so this little switch saves wear and tear on the plug and your sanity.

The cassette sense line is the next signal we have to process. In my recorder the remote plug is connected directly to the motor. This plug on most recorders is connected in the same way; it is usually in series with the motor. When the recorder is on, the cassette's voltage is present on these leads. When the recorder is off, one or the other of these leads is usually connected to ground through the motor.

Since that voltage will be unknown in many recorders, and since that voltage has to be inverted, we tie it through a resistor and diode to the +5 volts. This protects the input of the inverter which can only tolerate five volts. When there is no voltage, that is, a low at the input of the inverter, its output

is high. The computer sees this as a signal that indicates the recorder is off. When the recorder is on, that is, the play button is pressed, the voltage from the recorder causes the output of the inverter to go low, thus signaling the computer that the recorder is on.

There are two problems associated with your own recorder. The first is how to determine which line from the remote plug to use. This can be done by measuring the voltage with an open subminiature phone plug connected to the remote jack. When the recorder is on, one of the pins will have voltage on it. You want to use the other line.

The other problem is if the remote jack is connected between the motor and ground rather than between the drive voltage and the motor. If your own cassette recorder works like this, all is not lost. You can do one of two things. One is to rewire the jack, which may or may not be an involved job. The other is that you just have to be more aware of what is going on. You permanently tie the sense line to ground, and when you are ready to save a program, you just have to make sure that you press the record and play buttons before you press the <RETURN> key on the computer. And when you load a program, you have to press the <RETURN> key before you press the play button.

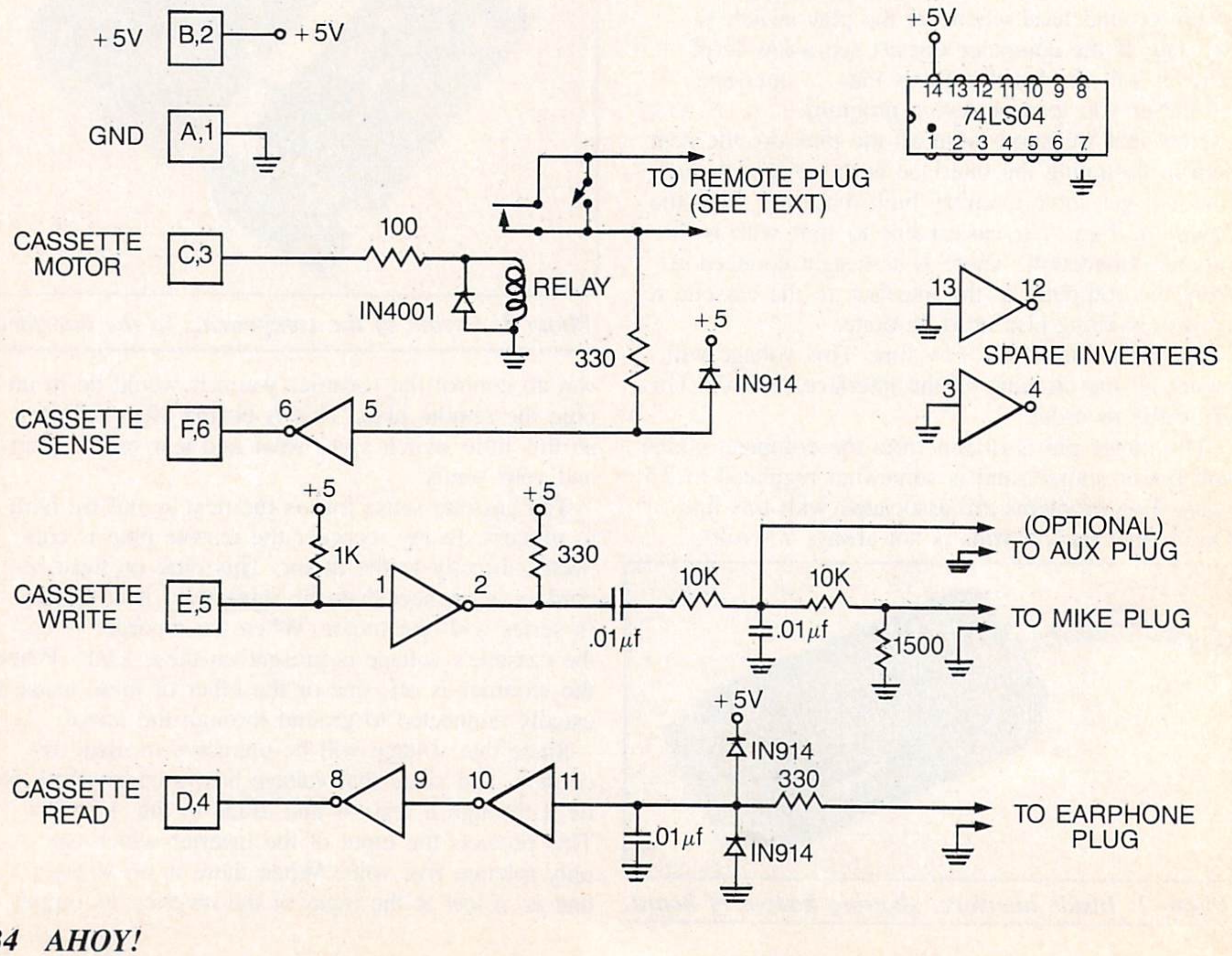
The next line we design for is the Cassette Write line. The computer sends out a signal that is to be saved onto tape. This signal is always five volts high, but varies in frequency. A zero bit and a one bit are converted into two different tones. I won't go into the theory of how or why this is done at this time, but if you play back the tape on the recorder, you can hear the varying tones. Of course you can't do this on a Datasette since it has no speaker.

The signal from the computer is buffered by one of the inverters. This is then filtered and reduced to a level that can be accepted by the microphone input of the recorder. Most recorders have a MIKE input, but not all of them have a high level (or LINE or AUX) input. If you want to use a high level input into to your recorder, there is a 1 volt signal available at the optional AUX output of the circuit (see schematic). Also, the automatic gain control of the recorder helps to maintain this signal at the right level for recording.

And finally the circuit for the Cassette Read line is designed. We must take the signal from the cas-

Continued on page 75

FIG. 1: SCHEMATIC OF CASSETTE INTERFACE





Chill out with help from Synapse.
READER SERVICE NO. 130

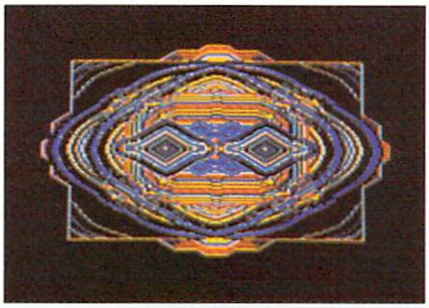
RELAX
Synapse Software
Commodore 64
Disk

Today's society is more competitive than any other in history. The hustle and bustle creates an atmosphere of tension that completely permeates our lives. Learning to control stress has become very important, leading people to meditation and biofeedback. Now the personal computer has entered the arena!

Relax from Synapse is a hardware/software package which combines biofeedback, autogenic training, and progressive relaxation techniques. Nervous already? Don't be. I'll walk you through my first encounter with the program.

I sat down at my C-64 and looked at the *Relax* booklet. It explained that *Relax* works by monitoring the frontallis muscle, the area around the temples (you can feel this muscle by clenching your teeth). Waiting for the disk to boot, I adjusted the special headband on my forehead. This connected to a module which in turn went into the Commodore's peripheral port. A graph popped up on the screen and I waited for a jagged line to show whether I was tense or not. But the line hugged the bottom of the screen. This relaxed I didn't want to be—it meant I was dead! I readjusted the headband and made certain that its sensors were making contact with my temples. This was better?! Now the jagged line was fluctuating like a fire hose gone crazy.

This teaches me to skim through instructions. An averaged rate has to first be set when using the program. This is done with a series of slide controls on the module. Then a fast or slow sample is set up. The fast one indicates how the unconscious is reacting while a slow sample gives a more general index of overall relaxation.



A graphic representation of stress.

The jagged line now averaged out, moving up and down as the graph scrolled from left to right. I dumped my results to a printer and consulted the manual. I found out I wasn't as hyper as I thought.

There are also three tension/relaxation games. *Sensorial Kaleidoscope* creates duplicating patterns at the edge of the screen. They increase in size as tension is reduced, moving towards the center. The balloon program has you navigating a hot-air ship across a scrolling landscape. The balloon picks up speed as you relax.

The final program uses a cartoon face to mirror your tension level. I was doing fine until I glanced at my watch and saw that I was late for another appointment. My happy face developed a most sour look.

What did I learn? Stress, unfortunately, is very much a part of our lives. Controlling it is not just beneficial—it's necessary. *Relax* ($149.95) has the potential for becoming not just "another" program, but a useful tool that few will want to be without.

Synapse Software, 5221 Central Avenue, Richmond, CA 94804 (phone: 415-527-7751).

—Marshal Rosenthal



Dig for black gold in **Oil Barons.**
READER SERVICE NO. 131

OIL BARONS
Epyx Computer Software
Commodore 64
Disk; keyboard

Unplug your joystick. Relax those catlike arcade reflexes. *Oil Barons* is a simulation that lets you play the role of a wildcat driller. Your goal, whether playing solo or with up to seven opponents, is to buy land at auction, survey and drill the best sites, and pump black gold straight into your bank account. Collect the most dough and you win.

Oil Barons is half board game. It comes packaged like an Avalon Hill bookshelf game and includes a large mapboard, a two-sided disk, and several hundred markers. Divided into 2000 squares, the map consists of small pictures of

thirteen different types of terrain from deserts to ice packs and cities to jungles. The markers are used to indicate such things as ownership, wells, dry holes, and national parks.

The computer plays several critical roles in *Oil Barons;* it is auctioneer, competitor, banker, realtor, Mother Nature, and newspaper publisher all rolled into one. There is no paper—money or score cards—in this game. You will find, however, that you quickly realize the need to keep detailed notes even on information that the computer stores. That information, like your debts and income or the results of oil surveys, is only available at a certain point in your turn. You will want to write it down so you can consider your strategy between turns.

As you set the game up, you can pick any of seven options for play. In Reality games, oil pools are generated under the playfield regardless of terrain types. In Classic games, the chance of striking oil is tied to the type of terrain you are drilling on. Quick games leave out or shorten various parts of the gameplay; wells never run dry and government never levies taxes. Other game options allow you to customize the rules to your heart's content.

When you begin play, the computer gives each company (player) four parcels of land made up of two to twelve squares each. At the auction at the beginning of each turn, you have the chance of purchase more parcels and occasionally to buy land from another player.

During each turn, companies can survey as many sites as they wish and drill at one site. Before you do anything, you will get a detailed estimate of the cost for the survey, drilling rig, and labor. Surveys typically are not too bad, $100,000 or so. But drilling is expensive. You will have to swallow hard before drilling on a site that received a bad survey. When you hit a gusher, though, it is worth the cost. Gushers pay twice the profit of normally producing wells.

Survey a swamp, and you will see (in simple graphics) an airboat blow through the reeds, planting a charge and a recorder. Then the screen will show a graphic readout of the detonation results. Drill in the ocean and a floating derrick will be built. Graphics for other terrain types are simple but appropriate.

At the end of each company's part of its turn, several screen printouts are shown including a portfolio of wells, income statement, balance sheet, and financial summary. At the end of each complete turn, the computer-published "Oil Street Journal" is printed onscreen. It ranks the competing companies, tells you which wells have run dry, and reports on Congressional action allowing National Parks to be surveyed.

Between turns you have the option of saving the game; you'll use it. Although a Quick game may be completed in a few hours, a full-featured, multiplayer game could run weeks or months. *Oil Barons* is a complex game. It is not difficult to play; in fact the instructions cover every detail in only ten pages. The computer really runs the game and prompts you through each turn.

The computer also slows things down. Each player, as his company surveys, drills, and gets reports, will need to sit at the computer for several minutes. Other players can use this time to strategize, but things can get pretty slow. *Oil Barons*, for all its complexity, may not keep you entertained while you are waiting.

Epyx Computer Software, 1043 Kiel Court, Sunnyvale, CA 94089 (phone: 408-745-0700).

—Richard Herring

LP-10S LIGHT PEN
Tech-Sketch Inc.
C-64, VIC 20

Light pens are a magical peripheral. Few things I do on my computer delight me so much as drawing with such a pen. Colors and lines bloom on your screen with the flick of a pushbutton switch on the barrel of the Tech Sketch Light Pen, Model LP-10S ($39.95).

As low-cost light pens go, the Tech Sketch is fairly accurate and easy to use. Plug it into the joystick port, load light pen software, and you're ready to go.

The Tech Sketch for the C-64 comes in a choice of two packages—with *Paint 'N' Sketch* or with *MicroIllustrator*. Both are drawing programs. The *MicroIllustrator* package is $30 more—$69.95. You can buy *MicroIllustrator* separately for $40, but *Paint 'N' Sketch* is only available with the pan. Tech Sketch also publishes a number of educational programs which use the light pen, some of which are very good. The LP-10S works on the VIC 20, although the VIC version comes with a different drawing program.



READER SERVICE NO. 176

If I did not have *MicroIllustrator*, I would probably be delighted with *Paint 'N' Sketch*. The latter is an elementary program. It features three colors (orange, green, and blue) and several drawing modes. You can do freehand drawing, lines, rectangles, triangles, and circles. A fill mode floods line-enclosed spaces with color, and erase lets you remove what you don't want.

Using the pen takes some practice. Small children may find it frustrating. Sometimes it's out of calibration and the cursor symbol on the screen is not directly under the point of the light pen. If the cursor is somewhere different than where you think it is, or if you don't hold the pen square to the screen, the computer may not read the signal.

When you choose a new color or a different drawing mode from the drawing menu, a bell rings to tell you that you succeeded. You may have to turn up the brightness or contrast to get the pen to respond reliably. If you use any drawing mode but freehand, you must choose it again after each operation. Having to select line mode for every straight line you draw is annoying.

While *Paint 'N' Sketch* is supposedly more elementary than *MicroIllustrator,* I found the latter easier to use. It also has much greater graphic capability. You have all 16 of the C-64's colors, plus two cross-hatch textures for

ATTENTION PROGRAMMERS!

***Ahoy!* is constantly in search of the best C-64 and VIC 20 programs available, in all categories: utilities, games, education, graphics, music, etc.**

If you've written a program of exceptional quality, send it to us on disk or tape, accompanied by a printout and a self-addressed envelope with sufficient return postage affixed. (Specify C-64 or VIC 20, and how much memory expansion, if any, is required.) *Ahoy!* pays competitive rates, on acceptance.

each color. You have a choice of several "brush" widths. You have lines, circles, filled-in discs and blocks, and rectangles as well as draw and fill.

The real fun begins if you choose mirror or rays. Mirror duplicates every line you draw so that four lines, which mirror one another, all appear as if by magic. When you select rays, you define two points, and a graceful fan of lines follows the arc of your pen. Other functions on the *MicroIllustrator* menu include magnify (which lets you work on a blown-up section of your picture), calibrate, erase, and disk operations.

Both *Paint 'N' Sketch* and *MicroIllustrator* let you save pictures to disk and reload them for viewing or change. I understand that new versions of *MicroIllustrator* will also have a printer routine for hardcopy versions of your pictures. Printing won't transfer the colors, but will let you use the program to print black and white designs.

Even *Paint 'N' Sketch* lets you create interesting pictures. *MicroIllustrator* gives you control and flexibility equivalent to almost any other two-dimensional art medium. I showed this program to an art teacher (who dislikes computers), and she was thrilled and amazed at how easily she could create spectacular effects.

Erasing part of the picture is difficult, as the erase menu undoes your most recent operation or blows away the whole picture. Drawing with the background color works for erase, but takes patience and skill.

Light pens and programs for them are likely to be increasingly available. The Tech Sketch pen works with other vendors' software, so you will not be locked into the Tech Sketch software line if you purchase it.

Writing your own programs for a light pen is easy. Either the user's manual for the light pen or the *C-64 Programmer's Reference Manual* has the essential information. Points around the edge of the screen are hard to use precisely, but that's your Commodore computer rather than the light pen.

If you want to experiment with a light pen, the Tech Sketch is as good as any in its price range. The trigger button on its barrel makes it easy to control, and is preferable to using a key on the keyboard, as the Edumate Light Pen does.

If you have dreams of drawing in light and color, I urge you to consider buying the more expensive Tech Sketch package with *MicroIllustrator*. You will soon outgrow the capabilities of *Paint 'N' Sketch*, but *MicroIllustrator* will delight your artistic ego for a long time. Save your masterpieces to disk, and you can display them at parties or send them to your friends or just look at them in fresh, living splendor for years after you make them.

Tech-Sketch Inc., 26 Just Road, Fairfield, NJ 07006 (phone: 201-227-7724).

—Annette Hinshaw

ESP>CALC

New Leaf, Inc.
C-64, VIC 20
Disk

No, this program won't help you solve complicated math problems through extrasensory perception. The ESP in this case stands for Electronic Spreadsheet Planning. Like most spreadsheets, *ESP>Calc* lets you perform a number of mathematical operations on several numerical values at once. The numbers, which are arranged in rows and columns, may be listed, added, subtracted, multiplied, divided, compared, and even forecasted for future periods.

If you've never used a spreadsheet, its exact usefulness may still seem somewhat unclear. This is partly because a spreadsheet is really a multipurpose piece of software. Depending on the spreadsheet's layout, the numerical data you supply it with, and the operations it's instructed to perform, a spreadsheet can help you plan your household budget, chart biorhythms, or even analyze the performance of selected stocks.

To help you get the most from *ESP>Calc*, the folks at New Leaf have packed the disk in a loose-leaf binder containing close to 150 pages of instructions, examples, and trouble-shooting tips. There's even an operations glossary that shows the exact formats to use for

each possible operation and illustrative samples that you can try for yourself. It seems like a lot of material to wade through, but only the first few sections are needed to get you started. In addition, most of the screens are self-explanatory and you always have the option of calling up the HELP screen to get a list of your available options.

To design your own spreadsheet you should first do a rough sketch on paper to figure out how many rows and columns you will need, their titles, and what operations you'll want it to perform. When you run the program it will ask you to input the number of rows and columns and then ask you the names of each. If you make a mistake or the spreadsheet has to be expanded at some later time, it is a simple matter to change the headings of columns and rows and even insert new ones as they're needed.

After the program checks for duplicate titles, it's time to specify the operations you want performed. Using a simple example from the instruction manual, if COL 1 represents the amounts you've budgeted for a given month, COL 2 the amounts you actually spent, and you want COL 3 to display the difference between the two, you would enter the following: C3=C1−C2. In the same example, ROW 4 receives the totals for all expenditures in the first three columns as a result of the operation R4=R1#R3. The "#" (number) symbol is used in *ESP>Calc* to mean the sum of the first row or column listed, through the last. In this case, ROW 1 through ROW 3.

Don't get the wrong impression. *ESP>Calc* can handle a lot more than simple household budgets. The manual also includes a completed model of a spreadsheet that can be used to monitor your utility costs. It contains 17 columns and 31 rows for a total of 527 individual cells. It performs 23 operations which calculate everything from changes in utility rates to total utility costs on a monthly, quarterly, and annual basis.

By studying this example and others included in the *ESP>Calc* manual, you will soon learn how to set up complex spreadsheets of your own. Whether you need a spreadsheet to calculate tax deductions or the odds for major sporting events, *ESP>Calc* can help you keep those numbers in line.

New Leaf, Inc., 120 Lynnhaven, Belleville, IL 62223.

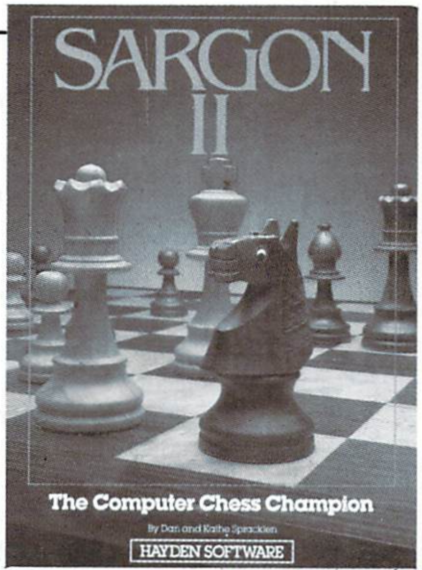
—Lloyd Davies

SARGON II

Hayden Software
C-64
Disk, cassette

If you have an ego, don't buy this game. Who wants to get stomped by silicon chips? *Sargon II* ($34.95) is an improved version of the original *Sargon;* it is both smarter and faster. So fast, in fact, that you will be tempted to play at higher skill levels. Where some chess programs will take hours to move at intermediate levels, *Sargon II* will respond in minutes. *(Sargon III,* running on an accelerated Apple II, was the first microcomputer program to beat a rated Chess Master in tournament play.)

Seven skill levels are available, numbered (as is the norm in the computer world) 0 through 6. The number of the skill level represents the minimum number of half moves ahead which the computer will consider. At level 0, where the computer considers only its current move, the response is immediate. By skill level 4, where the computer is considering its next three moves and your next



Seven skill levels are available.
READER SERVICE NO. 135

two, response time runs about six minutes. By level 6, the computer will take about four hours to consider the possibilities.

The chess board is conventional. You get a two-dimensional aerial view of the board with a profile view of each piece. Squares are represented by file (columns A to H) and rank (rows 1 to 8). Ranks are always labeled on both sides of the board. Files are another matter—an annoying matter. The line below the board shows your level, the number of half moves ahead the computer is thinking, and your move as you type it in. The file labels are not visible unless you toggle off the command line, and then you cannot see your moves as you type them in.

Sargon II has no joystick option; you must enter moves in algebraic notation (B1-C3 moves your knight out). If you enter an illegal move, or are placed in check, the computer beeps a warning. Any time you want to review the moves so far, a single keystroke will toggle to a second screen where both sides' moves are listed. All legal moves, includ-

ing castle and capture *en passant*, are allowed.

One of the best features of *Sargon II* is the ease with which it allows you to set up a midgame board. After moving to the square you want (with the cursor keys) just type the first letters of the piece's name and of its color. You can even indicate if it has been moved or not so the computer can determine the legality of certain moves. When you have the board suitably stacked for a sure win, tell the program which color moves next, what color and level you want, and approximately how many moves into the game your board is. The computer's strategy gets tougher toward the end of the game.

This editing feature can be used not only to set up a game, but also while you are playing. By going into edit mode, you can take back a move or two. Didn't notice that your queen was open when you advanced that pawn, did you? If you edit during the middle of a game, the second screen list of moves is not available. So if you are going to undo many moves, you may have to look at that screen and write everything down before you edit.

If *Sargon II* is proving to be too tough an opponent, you can always kibbutz. Press a function key and *Sargon* will offer up a possible solution to your predicament. Unfortunately for beginners, hints are not available at the lowest level. And don't feel that getting a little help is cheating. After all, the instructions ask, "Ah, but will *Sargon II* give you a move it doesn't have an answer for?"

Heyden Software, 600 Suffolk Street, Lowell, MA 01853 (phone: 800-343-1218).

—Richard Herring

WIZARD OF WORDS
Advanced Ideas
Commodore 64
Disk

Wizard of Words ($39.95) is a set of five educational games geared toward spelling and recalling words. On the front of the disk are the five games, on the back is the Wizard's Royal Registry of Words, a 38,000 word list. I wish the spelling checker for my word processor had as complete a dictionary! From the program's main menu, you select any game or the option of building your own word lists. Once that game is loaded, turn the disk over, or insert a copy of the word list, and you are off. If you're worried about changing disks in the middle of a program, don't be. Try to play a game without inserting the word list and the program responds "Oops! Disk problem: Be sure the *Wizard of Words* Disk is in place and that the disk door is closed." Each game has multiple difficulty levels, so this software is appropriate to a broad age range. The instructions say 7 to 13, but at my house a score of people from 5 to 38 have enjoyed the program.

In *Jester's Jumble* a medieval court jester with a two-pointed hat stands next to a striped circus tent juggling colored balls. When you are ready to play, he tosses the balls to the top of the screen where they freeze into the letters of a scrambled word which you must unscramble. After each incorrect guess, the jester will juggle the letters again and throw them up into a pattern that is a little easier to recognize. Eventually, he may solve the word before you do. The scrambled word is presented in large letters and the jester stops juggling while you are trying to unscramble them so you are not distracted.

The second game is *Castle Capers*, a non-violent version of hangman. Up to eight banners are rolled up along the escarpment of a stone castle. In the blue waters of the moat below float all the letters of the alphabet. When you guess a letter, it disappears from the moat so you are only prompted to guess (and the game will only accept) letters which have not already been chosen. If your guess was correct, the king runs out of the tower to one or more of the banners and unfurls your letter. If you are wrong, the castle gate lowers a little. When the gate is completely closed, you are out of guesses.

My favorite game on this disk is *Dragon's Spell*. Because I think it is the best game for kids? Nope, because it is the most fun for adults. A big green dragon with a pointed tail referees this game for two players (or two teams). After

he snorts out the letters of a long word onto the screen, he challenges you to make up as many shorter words as possible using only the letters of the long word. As you type in words, the dragon will peruse his little blue dictionary, his eyes rolling back and forth, looking for your word. If he finds it, he will shoot flame from his snout, your word will appear on a list, and you will get one point for each letter. (Only twelve words can show onscreen at one time.) If your word has already been used, he will shake his head and let you try again. Whenever you use a word that the dragon cannot find, he will tell you so and ask you to look it up in your dictionary. This friendly dragon has unquestioning faith when you tell him that your guess was really a word.

We have probably all played a game like this on the back of a paper placemat at Pizza Hut. The strength of *Dragon's Spell* lies in its interactivity. This dragon has the personality of the waving diplomats in *Choplifter* or the hero *Shamus*. Grownups can play with kids by changing a rule or two. The grownup can limit himself to six-letter or longer words and then cut his score in half. I especially appreciate this game's emphasis on using a dictionary. It's unusual for that particular book to get much use in a playful, fun activity. When the game ends, our pal the dragon says thanks for playing.

Word Spinning is the fourth game. A princess spins out a pattern on the screen similar to a crossword puzzle. Players take turns fitting words into rows or columns of blank spaces. The challenge comes in trying to think of a word when several of the letters (from intersecting words) are already filled in. When you type a word, the princess will check to see if it is in the Wizard's word list or ask you to check your dictionary. Each correct word increases your score. When the pattern is all filled in, a frog will hop across the screen, kiss the princess and, lo and behold—turn her into a frog. *Word Spinning* not only challenges you to spell words correctly, but also forces you to be creative in thinking up words which match the pattern.

The last game is *Herald's Hark*, a takeoff on *MasterMind*. You guess any word with the correct number of letters and, of course, on the Wizard's list. A herald will raise his horn and play a few notes which will turn into your word as they hit a tapestry in the middle of the screen. The number of correct letters will appear alongside your guess. Although separate word lists are displayed for two players, when the correct word is guessed, both heralds will play and the program will say "You guessed our word." Since both players' guesses provide clues about the hidden word, it seems appropriate that both players are rewarded.

Writing reviews is not always easy. I want to be fair to the software, but even more, I want to give an objective evaluation. *Wizard of Words* is one of those special programs that is easy to recommend. If you have elementary to junior high school aged kids, and especially if you would like to be able to play an educational game with them (and have fun yourself), this is a great package. Bright colorful graphics, a focus on the problem, freedom from bugs, and sound educational design are all rolled into this software.

Advanced Ideas, 2250 Ninth Street, Suite 104, Berkeley, CA 94710 (phone: 415-526-9100).

—Richard Herring

CREATING YOUR OWN GAMES

Continued from page 20

but they also have a lot in common with fiction. As an avid reader since childhood and an even more avid writer of fiction since they started paying me for doing it, I enjoy and admire a good story told well.

Part of telling a good story is creating an interesting, believable milieu, the environment where the story takes place. It is quite possible, as many adventure writers have proved, to create text adventures that are utterly dull, often because the world of the game is uninteresting, filled with humdrum items and people so boring you'd rather die than meet them at a party.

Many adventure game makers think that to make places interesting, they must describe them in purple prose: "You see a fantastically beautiful reddish-gold cliff. From the top of it flows a magnificent silver waterfall which creates a rainbowed mist at the base of the cliff." I mean, wow.

What makes a person, place, or object interesting is its potential for conflict, experience, or information. Too many games (including, alas, even *Zork*) are filled with objects that just sit there. A big emerald, for instance. To make it really exciting, they call it a *really* big emerald, or even a *huge* emerald. Almost too much excitement to bear, isn't it?

I much prefer finding a rusty knife in the hand of a skeleton, which turns out to have a curse on it so that the more you use it, the more likely you are to kill yourself with it. I prefer entering an ancient buried temple by climbing down a rope, or finding at the bottom of a stairway that I am standing at the gates of Hell. These all happen in *Zork I*. Surprises, yes, but they make sense in the context of the game; they are also memorable experiences.

A good adventure will have much the same effect as a good story. The author shapes the player's experience, but with computer adventures, the order of events and many key decisions depend on the player's own choices. There isn't much character development and dialogue is always a little weak, but there's plenty of action and suspense, and as many puzzles to solve as in a good mystery.

OTHER PEOPLE IN ADVENTURE GAMES

When I meet a person in my adventure, I want him to be doing something. Just standing with his hands in his pockets, whistling. Or skinning a rabbit. Or passing her hands over a crystal ball, muttering something under her breath while she surreptitiously glances in my direction. Or offering me a cup of what looks like hot milk. Or locking the door of the bank and getting nasty when I keep trying to get in.

Unfortunately, few adventure games really offer a chance to meet other characters and interact with them. It's hard enough to program the player's movements through a game without also having to worry about a dozen other characters.

The one adventure game that handles people superbly, however, is *Murder on the Zinderneuf* from Electronic Arts. In *Zinderneuf,* you are a detective who has to identify a murder on a luxury zeppelin flight before the airship lands. In the process, you must examine different rooms, talk to many people, and finally make accusations.

Each time you play, there is a different victim, a different murderer, and you yourself play the role of a different detective. You can choose from a list of several detectives, which are designed to resemble such fictional stalwarts as Hercule Poirot and Miss Marple. Naturally, suspects will have differing reactions to the different detectives. Also, you can choose the *manner* of your questioning. If you happen to be a good-looking woman detective, you can choose to be seductive; if you are a physically imposing man, you can choose to be belligerent.

This makes for fascinating, unpredictable interactions between you and the various suspects. Most adventure games run the risk of becoming dull because after a few plays you've seen most of the scenery; *Zinderneuf* always has the same scenery, but the people create many possibilities. It makes it a superb storytelling game—which is no surprise, since one of the creators is a sometime science fiction writer, besides being an excellent game designer. I'm never surprised when science fiction writers turn out to be brilliant in other fields as well...

PICTURES AND PUZZLES

Murder on the Zinderneuf doesn't confine itself to words—it takes place on a floorplan of the zeppelin, and you move around like a figure in a videogame, as do the other characters. Words only take up a small part of the screen. This graphic approach has been taken by other game designers.

Many games show pictures of each scene in the game, instead of using verbal descriptions. Some pictures are quite sketchy—virtually diagrams. Other games use pictures that are as close to cartoon animation as is possible on a home computer. But a more significant difference, I think, is the way the pictures are used.

If the pictures are used like still photographs, with

all the action taking place using words that appear in one part of the screen, I consider it to be an *illustrated text adventure.* A good illustrated adventure will still convey vital information through the pictures—you should be able to learn things by looking, rather than just seeing a pretty scene.

If the action actually takes place inside the picture—if figures move around and interact, almost like a videogame, even though words might also be used, then I call it an *animated adventure.* Animated adventures begin to act like arcade-style videogames, except that they still use the "room to room" pattern of text adventures, so that you explore, find objects, and interact with other characters. Unlike videogames, you have times when you can pause to think, to reflect, to figure things out, though at other times you might have to move quickly to save your life.

PROS AND CONS OF THE DIFFERENT APPROACHES

All adventure games use up memory in big gulps. Figure that a good adventure game will probably have at least 30 different rooms or scenes—most have many times that number. Each scene will have a description, which consists of many words. And there'll be at least a dozen objects, and a few characters, and many, many different events that can happen, all of which have to be included in the game. There aren't many adventures that can fit in an unexpanded VIC. My very simple *Emerald Elephant of Cipangu,* the same game that you'll see in its entirety next month, won't fit in the VIC—even without the REM statements it requires approximately 13K of memory.

Adventure gamewrights aren't alone in worrying about memory. Memory is always the limitation on the game designer's creativity. But it's an especially knotty problem with illustrated and animated adventures: If each screen display takes 8K on the high-res screen, how many different screens can you fit on each disk? How many can reside in memory at a time? Each time the player moves to a place whose illustration isn't already in RAM, the program has to stop and load the 8K display from disk, while the player waits, impatient to see what happens next.

Text adventures have it easier—but even they usually resort to disk accesses during the course of a game, because not all the game will fit into memory at once. Each disk access slows down the game, especially with a drive as slow as the 1541, and since all-text adventures usually don't have to access the disk as often, text adventures are usually much faster to play than the graphics adventures.

Infocom claims, and I tend to agree with them, that text adventures have another advantage over graphics adventures and videogames besides memory use and speed. It is the same adventure that books have over movies. As a writer of books, I admit I am biased, but the fact remains that in my science fiction novels I can create special effects that would cost George Lucas's Industrial Light and Magic, Inc., millions of dollars to bring off—if they could do it at all. And all it costs me is a few keypresses on my word processor.

Similarly, an all-text adventure shows you things that could never exist anywhere but in imagination—because it is in your imagination where the scene is drawn. A good artist for an illustrated adventure might be able to show you an excellent version of that place—but it will be only one artist's version, and only one angle of view, and it will only look as good as the computer itself will allow. To my way of thinking, you and I can still imagine much better art than any computer artist has yet shown us.

At the same time, this is also text adventures' disadvantage: there's nothing to see on the screen but words.

Animated adventures have an advantage over text or illustrated adventures. When you actually control player movement with a joystick, it becomes more immediate; players experience the events more directly than when they have to type in words, especially if they aren't 60 wpm touch typists.

Someday, when home computers routinely use videodisc technology to access thousands of screen displays in an instant, with very, very high-resolution graphics, we'll see adventures that combine all the advantages of text and graphics adventures, along with Disney quality cartoon drawings. For now, though, I'll stick with pure text adventures.

The nicest thing is that text-only adventures really aren't hard to program. They're hard to *create,* in the same way that a short story is hard to create—but the actual programming, once you know how tables work, is pretty straightforward.

SO LET'S START PROGRAMMING

In this column we'll create the basic structure of a small text adventure game. Next month we'll flesh it out and create a complete—but brief—game called *The Emerald Elephant of Cipangu.*

Where do we begin? Let's set this adventure in a castle. In the tradition of most adventure games, we'll begin outside the castle so the player has to figure out how to get inside. Figure 1 is a rough floorplan of the castle, with an elevation to show

how the upper and lower levels stack up.

If this were an animated adventure, our programming task would be to draw the castle and allow a figure to move around in it, something we've done in past installments of this column. But a text adventure uses a completely different method. Instead of putting this floorplan on the screen, we divide it up into a Room and Direction Table.

This floorplan naturally divides itself into 13 rooms. A "room" doesn't mean an indoor space—it refers to any one place where the player can be. So there are four rooms outside the castle: the area north of the castle, west of the castle, east of the castle, and south of the castle. There are 9 rooms inside: the kitchen, the courtyard, the gatehouse, the walls where the castle defenders would stand, the great hall, the bedchamber, the top of the tower, the stables, and the dungeons.

Naturally, a real castle would have many more possible locations—and we could as easily program a hundred rooms as fourteen. But the editors of *Ahoy!* assure me that they want to have room in the magazine for something besides my column, so we're keeping this very small.

Once we decide what rooms you want in your adventure, write them down in a particular order and number them. It doesn't matter what order they're in —it makes no different whether the great hall is room 5 or room 3.

These numbers, which represent the *order* of the rooms in the Room Direction Table, will be the key to the whole program. To keep track of where the player is at any one time, all we need to know is the number of the present room, which we'll store in the variable PR.

That number is the index into other tables. For instance, the Room Name Table, stored in the string array RN$*(n)*, will hold the room names in the same order as the Room Direction Table. To display the name of the present room, we just PRINT RN$(PR) and we automatically have the right name.

Sometimes there's a logical reason for grouping rooms together in a particular order. By numbering the rooms outside the castle from 1 to 4, we can test to see if the player is inside or outside the castle with a simple IF PR<5 THEN...or inside the castle with IF PR>4 THEN....

It isn't enough, however, just to number the rooms. Each room needs to have a sub-table with it—the list of what lies in each direction the player might move from that room.

Since one example is worth a thousand words, let's lay out the table. Here is the list of rooms in the order we'll use:

1. CASTLE MAIN GATE
2. MEADOW WEST OF CASTLE
3. LEDGE EAST OF CASTLE
4. GROVE SOUTH OF CASTLE
5. KITCHEN
6. COURTYARD
7. GATEHOUSE
8. ON THE WALLS
9. GREAT HALL
10. COUNT'S CHAMBER
11. TOWER LOOKOUT
12. STABLES
13. DUNGEON
14. TREASURE ROOM

The TREASURE ROOM wasn't on the floorplan. That's because we're going to assign its location randomly during the program, so that it can be in one of many different locations each time the game is played. That way the game will be different each time it's played.

We also need a Direction List. Later, this list will be part of the Command Table. Let's have eight possible directions: North, South, East, West, Up, Down, In, and Out. We'll number them accordingly:

1. NORTH
2. SOUTH
3. EAST
4. WEST
5. UP
6. DOWN
7. IN
8. OUT

These two lists are combined, so that for each room, there is a series of eight numbers, telling what room the player will reach if he goes in a particular direction. Let's look at the Direction Table for room 6, the courtyard:

Direction	Destination
1 (North)	7 (Gatehouse)
2 (South)	5 (Kitchen)
3 (East)	9 (Great Hall)
4 (West)	12 (Stables)
5 (Up)	7 (Gatehouse)
6 (Down)	5 (Kitchen)
7 (In)	9 (Great Hall)
8 (Out)	7 (Gatehouse)

Notice that three different directions, North, Up, and Out—lead to the Gatehouse. This is because the Gatehouse lies to the north, the courtyard slopes upward toward the Gatehouse, and when you want to go out, the Gatehouse is the logical place to go. It will often happen that two different direction instruc-

tions will lead to the same place.

The Room Direction Table will be set up in the two-dimensional array RD*(n,n)*. The first index is the room number; the second index is the direction number. For instance, suppose we are in room 6, the courtyard, and the player asks to go north. The player's command is held in the variable CM, which has a value of 1, which means *north*. The room we are presently in is PR. This statement will put us in the new room:

```
PR=RD(PR,CM)
```

Since PR=6 and CM=1, PR is now set to the value of RD(6,1), which has a value of 7. Therefore we are now in room 7, the Gatehouse. If the player had chosen to go *in*, CM would have had a value of 7, so PR would now be set to the value of RD(6,7), which has a value of 9. So we would be in room 9, the Great Hall.

COPING WITH ILLEGAL MOVES

Often the player will ask to go in a direction that doesn't lead anywhere. For instance, if the player is at Castle Main Gate, we want him to be able to go east or west, to get to either Meadow West of Castle or Ledge East of Castle. But we don't want him to go any other direction. After all, we aren't putting anything north of the castle at all, and it would be too obvious to let him get in the main gate. So we need to have some dummy rooms—rooms where you can't actually go, which allow the program to respond to the player's directions.

A simple solution is just to say that all illegal directions will return a value of 50, and the "name" of room 50 is "YOU CAN'T GO THAT WAY." Then, if the player called for an illegal direction—say, RD(1,1), which means "go north from Castle Main Gate"—the value of RD(1,1) would be 50. When the program PRINTs the new room "name," it would print RN$(50), which says "YOU CAN'T GO THAT WAY."

An easy way to enhance a game, however, is to have several different "illegal move" messages. For instance, if the player tries to climb the castle walls by commanding *up*, that direction can have a value of 51, which says, "YOU CAN'T CLIMB IT." Or if the player ties to go off the cliff to the east of Ledge East of Castle, the value of RD(3,3) can be 54, and RN$(54) would say, "ARE YOU TRYING TO GET US KILLED?"

Why so many different illegal-move messages? For one thing, it makes the game more interesting. It gives the illusion that the computer is "alive," watching what's going on and responding differently for different moves.

Also, the illegal-move messages can communicate vital information. In the full game next month, we'll require the player to be holding certain objects in order to get through the secret door into the Treasure Room. If the player tries to go the direction that leads to the Treasure Room while not holding those objects, he'll get the error message "YOU DON'T HAVE WHAT IT TAKES TO GET IN THERE." That's a clue that if he *did* have the right things in his possession, he *could* get in.

Illegal-move messages can also imply that the game world is larger than the actual number of rooms would imply. For instance, this message hints that there are many other rooms in the castle: "YOU SEARCH THAT AREA AND FIND NOTHING OF INTEREST."

In our game we'll have six illegal-move messages, numbered from 50 to 56. Each will have an entry in the Room Name Table, so RN$*(n)* has to be DIMensioned to 56. However, since we will never actually *go* to those rooms, the Room Direction Table does not have to have an entry for those "rooms." RD*(n)* is therefore DIMensioned only for the total number of rooms where the player can go, which is 14.

SINGLE-KEYSTROKE PARSING

Instead of going into complicated parsing algorithms, our game will be controlled by the simplest possible interface. Every command can be entered by pressing a single key on the keyboard. This enables us to PEEK the player's instructions at location 197.

The Command Word Table, which will be contained in the string array CW$*(n)*, has the following entries, numbered from 0 to 16:

Number	Letter	Word
0	illegal	"WHAT?"
1	N	"NORTH"
2	S	"SOUTH"
3	E	"EAST"
4	W	"WEST"
5	U	"UP"
6	D	"DOWN"
7	I	"IN"
8	O	"OUT"
9	B	"BACK" (go back to the last room we were in)
10	T	"TAKE" (pick up an object in this room)
11	L	"LEAVE" (leave one object

		we're carrying)
12	P	"PEER" (peer closely at this room to learn more about it)
13	G	"GOT?" (what objects do we have right now?)
14	Q	"QUIT" (give up and quit the game)
15	H	"HELP" (display my options)
16	?	"HELP" (same as 15)

When the player presses one of the keys, the appropriate word will be displayed at the top of the screen, as if he had typed the entire word. This is the opposite approach to the one Infocom uses. Where Infocom allows the player to type sentences that are very much like English, in our game we're expecting the player to learn a code. Infocom's approach is much better—but much harder to program. We're going for simplicity right now, so we'll make the player memorize some commands and save ourselves the work of developing a parser.

EXPLORE THE CASTLE

We're just about ready to write the actual program. Remember, though, that this is only the simple, preliminary version. Only the directional commands will work. All the others will display on the screen, but nothing will happen. So all you can do with this program is move from room to room, getting the illegal move messages or the new room name.

Still, the Treasure Room is randomly assigned and you have to figure out where the castle can be entered. Also, the movement patterns are pretty tricky. You can't always get back the way you came. For instance, the door in the Great Hall that leads out into the Courtyard is in the north of the great hall. But if you're in the Courtyard, you are assumed to be standing in the middle of the open area, so that the Great Hall is to your *east.* To go from the Great Hall to the Courtyard, then, you would type N for *north;* to get back from the Courtyard to the Great Hall, however, you would type, not S for *south,* but E for *east.*

The program listing is found on page 94. I have included REM statements to identify what is going on, but as usual you don't need to type these in. They're only there to help you follow the program logic.

The program lines from 1000 on are the set-up routine. All the values for the various tables are contained in DATA statements, which I have tried to make as clear as possible.

The main loop of the program starts at line 100. It jumps to the subroutine at 990, which gets a keypress and converts it to a command number (CM); if the command is a directional command (1-8), the program jumps to the movement subroutine at 300; otherwise the main loop simply PRINTs the command word CW$(CM) and goes back for the next keypress.

Next month, we'll flesh out the program by implementing all the other commands, so the player can pick things up and put them down. Then, instead of just exploring the rooms of the castle, the player will be searching through the Castle of Darkness for the magic Emerald Elephant of Cipangu. The player will be able to open the main gate and unlock the door of the hidden Treasure Room—but only if he can figure out which objects he needs to be carrying in order to do it.

In the meantime, there's nothing to stop you from using this simple exploration program as the basis for your own game. You don't want a castle? Just change the room names in the Room Name Table. You want more rooms? Just DIMension RD*(n,n)* for the number of rooms you want and add entries to the Room Name Table and the Room Direction Table. For instance, if you want 30 rooms, you would DIMension RD(30,8). (Notice that the number of *directions per room* remains the same.) As long as you assign eight destinations for each room in the room direction table, you can handle any number of rooms up to 49 with this simple program. □

SEE PROGRAM LISTING ON PAGE 94



"Get thee behind me while I think it over."

COMMODARES

PROGRAMMING CHALLENGES

By Dale Rupert

Each month, we'll present several challenges designed to toggle the bits in your cerebral random access memory. We invite you to send your solutions to:

Commodares, c/o *AHOY!*
P.O. Box 723
Bethel, CT 06801

We will print and discuss the cleverest, simplest, shortest, most interesting and/or unusual solutions. Be sure to identify the *name* and *number* of the problems you are solving. Also show sample runs if possible, where appropriate.

Your original programming problems would be equally welcome!

Michael Russell (Mattoon, IL) pointed out that the simple password program described in the February *Rupert Report* would not be very difficult to break into and bypass. Of course, he's right. It was primarily an example of the non-printing characteristic of the GET statement. He asked if there is a way to make a program line unlistable. If you know of a way to do that, and you don't mind sharing your secret with the world, write and let us know.

In the meantime, Marc Spooner (Pembroke, Ontario) sent a more sophisticated protection program. That leads us to:

Problem #9-1: Passed Words

See how long it takes you to figure the password sequence required to be able to run the program which starts out with Marc's password protection scheme. The protected program would begin at line 130, and lines similar to line 140 would be sprinkled throughout, so you can't just break and GOTO 130 to bypass the protection. A word of caution: if you type this program into your computer, be sure to SAVE it *before* you run it. Otherwise, it may vanish before your very eyes!

```
1 :REM < COMMODARES #9-1 :
2 :REM      PASSED WORDS    >
3 :REM SUBMITTED BY MARC SPOONER
5 :GET O$ IF O$<>"" THEN 5
10 POKE53281,1:POKE53280,1:PRINTC
HR$(144)CHR$(147)CHR$(17)CHR$(17)
15 INPUT"PASSWORD:";A$
20 PRINT:PRINT:INPUT"PASSWORD:";B
$:Q=Q+1
25 S$=CHR$(82):IF MID$(B$,3,1)<>S
$ THEN NEW
30 PRINT:PRINT:INPUT"PASSWORD:";C
$:Q=Q+1
35 PRINT:PRINT:INPUT"PASSWORD:";D
$
40 L=LEN(A$)+LEN(B$)+LEN(C$)
45 L=L-LEN(D$):Q=Q+LEN(D$)
50 IF LEN(B$)<>L THEN NEW
60 IF RIGHT$(A$,1)<>RIGHT$(B$,1)
OR LEFT$(B$,1)<>MID$(D$,5,1) THEN
 NEW
70 IF A$<>MID$(D$,1,4)THEN NEW
80 IF LEFT$(C$,1)<>CHR$(77) OR MI
D$(D$,2,1)<>CHR$(65) THEN NEW
90 GET V$:IF V$="" THEN 90
95 GET X$:IF X$="" THEN 95
100 Z$=V$+X$:Q=Q+1
105 IF Z$<>MID$(D$,7,2) THEN NEW
110 IF V$<>MID$(A$,3,1) THEN NEW
120 IF Q<>LEN(B$)+3 THEN NEW
125 IF RIGHT$(C$,1)<>CHR$(67) THE
N NEW
130 PRINT"WELCOME TO THE PROGRAM!
"
140 IF Q<>LEN(B$)+3 THEN NEW
150 :REM PUT STATEMENTS LIKE LINE
 140 THROUGHOUT THE PROGRAM
```

Problem #9-2: Never Ending

This problem is based upon a program submitted by Haley Carter Jr. (Dayton, OH). Haley usually makes the Commodares harder for himself by trying to do them in assembly language. This program was written in BASIC.

Without typing it in, see if you can determine what will be printed on the screen when this program is run. And yes, line 40 is correct as printed. (See the March *Rupert Report* for a discussion of logical variables if this looks strange to you.) Since line 10 uses a STEP of 0, will this program ever end?

```
10 :REM < COMMODARES #9-2 :
11 :REM          NEVER ENDING? >
12 :REM SUBMITTED BY HALEY CARTER
```

```
JR.
20 FOR J=0 TO -1 STEP 0
30 N=N+1 : PRINT N;
40 J=N=50 : REM YES, J=N=50
50 NEXT J
60 END
```

Problem #9-3: Common Letters

The user types two words. The computer then tells how many of the letters in the first word are also contained within the second word. For example, if the user types "TEST,TAKE", the computer responds 3 since three of the letters of "TEST" are found in "TAKE". On the other hand, if the user types "TAKE,TEST", the computer responds 2 since only two of the letters in "TAKE" are contained in "TEST".

Problem #9-4: Letter Sorter

The user types a word or sentence, and the computer prints all the letters, including duplicates, in alphabetical order. For example, typing the sentence "COMPUTING IS FUN" would produce the output "CFGIIMNNOPSTUU". There will be as many letters in the output as there are in the original input.

This month we will look at several readers' solutions to March and April *Commodares*. We will wait until readers have had a chance to respond to last month's problems before we discuss them.

David Wharton (Gulport, MS) pointed out out that spaces and apostrophes prevent the famous palindrome "MADAM I'M ADAM" from being recognized by the simple palindrome solutions. His program included a test such as in line 20 below to eliminate spaces and apostrophes. Line 30 builds two strings, A$ and B$ which are opposites of each other (as seen by the order of the concatenation A$+Z$ and Z$+B$). If they're equal in line 40, then the original string X$ is a palindrome. You might modify further to eliminate other punctuation.

```
5 :REM SOLUTION TO COMMODARES
6 :REM PROBLEM #3-2 : PALINDROMES
7 :REM SUGGESTED BY DAVID WHARTON
8 :
10 PRINT CHR$(147):INPUT"WORD OR
WORDS";X$:L=LEN(X$)
20 Z$=MID$(X$,L,1):IF Z$=" " OR Z
$="'" THEN L=L-1:GOTO 20
30 A$=A$+Z$:B$=Z$+B$:L=L-1:IF L>0
 THEN 20
40 IF A$=B$ THEN PRINT"THIS IS A
PALINDROME"
```

A purely mathematical solution to *Problem #4-1: Squared Sum* as submitted by Daniel Amodeo (Bethesda, MD) is shown below. A few other readers also used this method of creating numbers digit by digit as shown in line 20. Most other solutions used strings to build numbers as shown in the May issue of *Ahoy!*

```
5 :REM SOLUTION TO PROBLEM #4-1 :
6 :REM         SQUARED SUM
7 :REM SUBMITTED BY DANIEL AMODEO
8 :
10 FORI=0 TO 4:FORJ=0 TO 9:FORK=0
 TO 9
20 V=100*I + 10*J + K
30 S=(I+J+K)*(I+J+K)
40 IF V=S THEN PRINT V
50 NEXT:NEXT:NEXT
```

The solutions of 0, 1, and 81 did not require any special testing such as IF ABS(V-S) <1E-4 in line 40, although it's a good idea to use such a test for larger numbers (see the *Rupert Report* in the April issue of *Ahoy!*).

Paul Dawson (Springhouse, PA) sent the following program for solving *Problem #4-2: Sum of Cubes.* The interesting thing about this program is that it lets you see the activity of the computer by means of the <Cursor UP> (CHR$(145)) in line 30. This program finds the solutions 0, 1, 153, 370, 371, and 407 also without special testing.

```
5 :REM PROBLEM #4-2 :
6 :REM   SUM OF CUBES
7 :REM SUBMITTED BY PAUL DAWSON
8 :
10 FORA=0TO9E8:T=0:FORB=2TOLEN(ST
R$(A)):V=VAL(MID$(STR$(A),B,1))
20 T=T+(V*V*V):NEXT:IFA=TTHENPRIN
TA"!"
30 PRINT A;CHR$(145):NEXT
```

The one-line solution to *Problem #4-3: Wrong Number* from Betty Colgin (Paxton, ME) is listed below. She indicates that she has worked with her computer for only a month, and that one reason for getting it was to help her "keep up with grandkids." Judging from this solution, she has a good start:

```
1 INPUT"LETTER";L$ : PRINT L$" IS
 LETTER NUMBER" ASC(L$)-ASC("@")
"IN THE ALPHABET"
```

Below are three variations on the theme of *Problem #4-4: Random Repetition.* Christopher Hebert (Sunnyvale, CA) mentioned that line 10 in his solution is a speedy way to fill the screen color memory with a value of 0. Clearing the screen on the C-64 (PRINT CHR$(147)) fills color memory which resides from 55296 to 56295 with whatever value is stored in location 53281. This is much faster than the FOR-NEXT loop:

```
FOR M=55296 TO 56295 : POKE M,0 :
NEXT
```

which does the same thing. Christopher and several other readers used the fact that a reversed character has a CHR$ value which is 128 greater than its non-reversed counterpart:

```
5 :REM SOLUTION TO PROBLEM #4-4 :
6 :REM     RANDOM REPETITION
7 :REM SUBMITTED BY CHRISTOPHER H
EBERT
8 :
10 POKE53281,0:PRINTCHR$(147):POK
E53281,1
20 FOR J=1024 TO 2023: Q=INT(RND(
0)*10)+48:POKE J,Q
30 IFQ=C THEN POKEJ-1,Q+128:POKE
J,Q+128
40 C=Q:NEXT
```

Paul Dawson (Springhouse, PA) took the problem one step further. He changed any two matching characters, either horizontally or vertically, to reverse video with the program shown below.

```
4 :
5 :REM PROBLEM #4-4 : RANDOM REPE
TITION
6 :REM SUBMITTED BY PAUL DAWSON
7 :
10 POKE 53281,1:PRINT CHR$(147):P
OKE 53281,6
20 FORA=1024TO2023:B=INT(RND(1)*1
0)+48:POKEA,B
30 FORX=1TO40STEP39:IFPEEK(A-X)=B
ORPEEK(A-X)=B+128THENPOKEA-X,B+12
8:POKEA,B+128
40 NEXTX,A:WAIT197,1
```

The WAIT statement at the end of his program halts the program until a key is pressed. (A problem for you experts is to figure out which keys will work with this statement.) The "current key pressed" buffer is at address 197. It is a handy way to keep the READY prompt from cluttering up your display until you're ready for it.

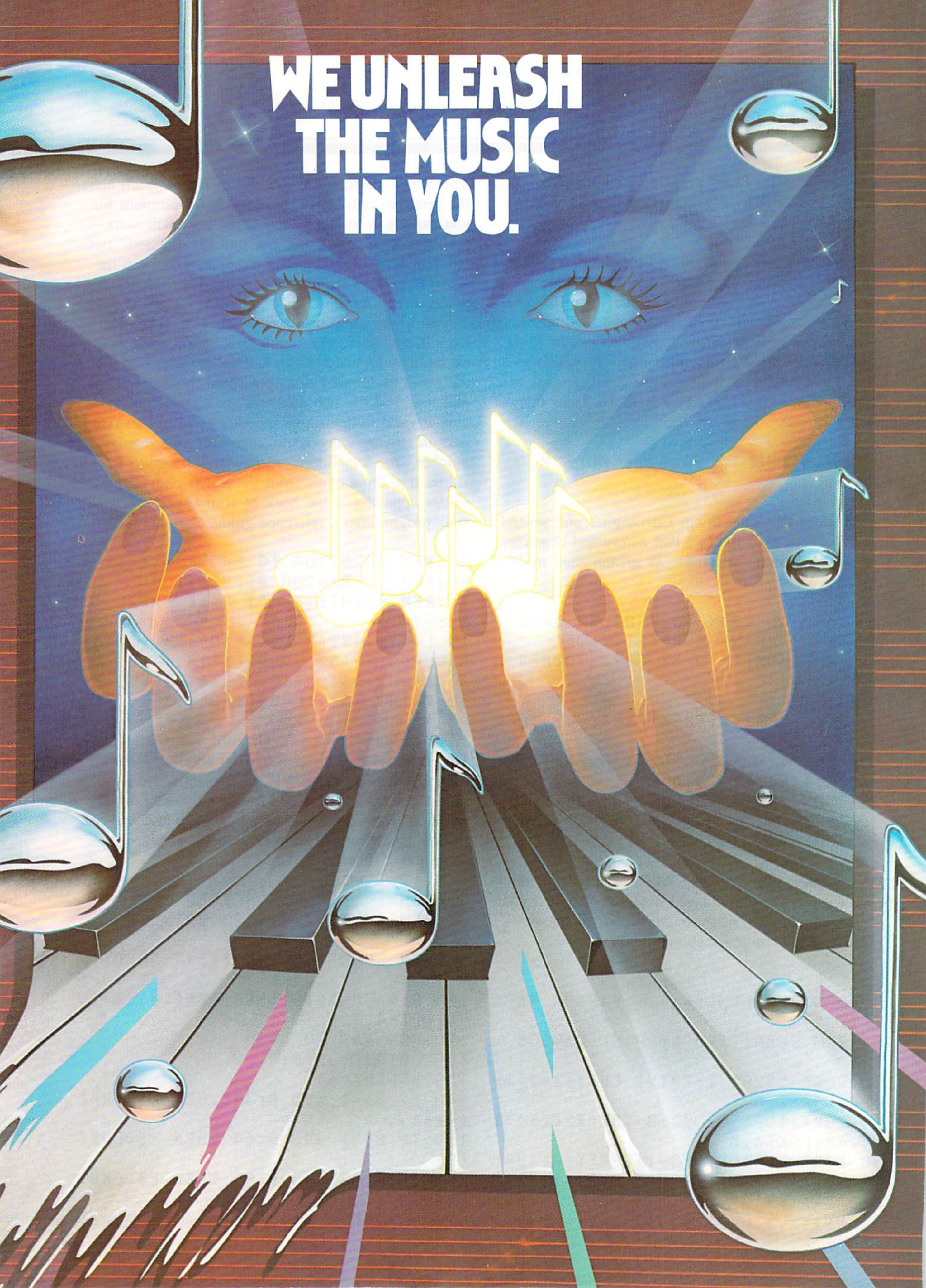
A third variation on *Problem #4-4* is shown below from Mark Nekic (Eastlake, OH). Mark's solution exemplifies logical variables. The variable T is 0 if the previous number does not equal the current one, and it is 1 if the two numbers are equal. This variable is used to determine whether to print two numbers in reverse video or to print only one regular number. You can figure out how this program works if you use some clues from the *ASCII and CHR$ Codes* appendix at the back of your user's manual or your programmer's reference guide.

```
5 :REM SOLUTION TO PROBLEM #4-4 :
6 :REM     RANDOM REPETITION
7 :REM SUBMITTED BY MARK NEKIC
8 :
10 L=X:X=INT(RND(1)*10):T=-(L=X):
B$=CHR$(T*157)+CHR$(T*18)
20 A$=B$+B$+CHR$(T*(L+48))
30 PRINT A$;X;CHR$(146);:GOTO 10
```

Brian Dobbs (Timmins, Ontario) sent the following *Low Resolution Copy* program for sending los-res graphics screens to the VIC-1525 printer. He says that it eliminates the space on the printout normally produced by carriage returns. If anyone else has reasonably short, useful utilities, this is the place to share them. Thanks, Brian.

```
5 :REM   < LOW RES SCREEN COPY
6 :REM     TO VIC-1525 PRINTER >
7 :
8 :REM SUBMITTED BY BRIAN DOBBS
9 :
100 SI$=CHR$(15):BS$=CHR$(8):D=10
24:OPEN4,4
110 FOR A=D TO D+39
120 PRINT#4,SI$;
130 B=PEEK(A)
140 IF B>-1 AND B<32 THEN E$=CHR$
(B+64)
150 IF B>31 AND B<64 THEN E$=CHR$
(B)
160 IF B>63 AND B<96 THEN E$=CHR$
(B+32)
```

Continued on page 76


WE UNLEASH THE MUSIC IN YOU.



Sight & Sound Music Software can do more than turn your Commodore 64 into a music synthesizer.

Deep within every Commodore 64 lurks the unexpected. A versatile music synthesizer. A music video machine. Even a three-track recorder. All waiting for you to control. All you need is Sight & Sound Music Software to make and record sounds that rival those of a real synthesizer. Including brass, strings, piano, guitar, electric bass, drums, funky clav, space wah and other sounds.

It's so easy that anyone can create and record music.

If you can press a few computer keys, you have all the skill it takes to create and play music. They're that music friendly. Our easy-to-follow, fully documented instructions guide you through every program.

And you can let your imagination soar. Creating and recording your own sounds. Your own songs. Your own music. Using the polyphonic keyboard. Playing across a full 8-octave range.

With one-finger, you can control wave form, vibrato, sweep and other special effects.

No one else offers our exciting animated graphics. Or the music of the famous recording artists we offer.

Every Sight & Sound Music Software program features eye popping graphics that explode across the screen.

And you can generate your own colorful graphics. Then, synchronize them to your own music...or to melodies made famous by Michael Jackson, Lionel Richie, Billy Joel and others.

A dozen different ways to have fun with computer music.

Sight & Sound Music Software offers music for everyone's musical interest. Like the Kawasaki Synthesizer that instantly puts a synthesizer's sound and flexibility at your fingertips.

Or, the Kawasaki• Rhythm Rocker that creates space sounds and graphics to the pre-recorded rhythms of electronic sound.

Or, even our Music Processor that allows you to create, edit, record, play and even compose your own musical arrangements.

From learning programs to the songs of your favorite artists...we've got it all.

A keyboard that makes easy-to-play even easier.

An easy way to get started making music is to begin with our Incredible Musical Keyboard. It fits over your computer keyboard and allows easy note-by-note play. Included with this optional keyboard is an introductory music software program and two music books. One for the beginner, the other for the more advanced musician.



So, see your local computer software dealer for all the exciting Sight & Sound Music Software. And unleash the music in you.

SIGHT & SOUND
Music Software, Inc.

We unleash the music in you.

Commodore 64 is a registered trademark of Commodore Business Machines, Inc.

Reader Service No. 166

3-84

SANYO MONITOR SALE!!



9" Data Monitor

- 80 Columns × 24 lines
- Green text display
- Easy to read - no eye strain
- Up front brightness control
- High resolution graphics
- Quick start - no preheating
- Regulated power supply
- Attractive metal cabinet
- UL and FCC approved

• *15 Day Free Trial - 90 Day Immediate Replacement Warranty*

9" Screen - Green Text Display	**$ 69.00**
12" Screen - Green Text Display (anti-reflective screen)	**$ 99.00**
12" Screen - Amber Text Display (anti-reflective screen)	**$ 99.00**
12" Screen-Super 1000 Line Amber Text Display	**$119.00**
14" Screen - Color Monitor **(High Resolution)**	**$239.00**

Display Monitors From Sanyo

With the need for computing power growing every day, Sanyo has stepped in to meet the demand with a whole new line of low cost, high quality data monitors. Designed for commercial and personal computer use. All models come with an array of features, including upfront brightness and contrast controls. The capacity 5 × 7 dot characters as the input is 24 lines of characters with up to 80 characters per line.

Equally important, all are built with Sanyo's commitment to technological excellence. In the world of Audio/Video, Sanyo is synonymous with reliability and performance. And Sanyo quality is reflected in our reputation. Unlike some suppliers, Sanyo designs, manufactures and tests virtually all the parts that go into our products, from cameras to stereos. That's an assurance not everybody can give you!

SANYO

Official Video Products of the Los Angeles 1984 Olympics

• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. Visa - MasterCard - C.O.D.

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

COMMODORE-64 or VIC-20

VOICE SYNTHESIZER

NEW

NEW

MAKE YOUR COMPUTER TALK

VOTRAX BASED HARDWARE

SALE

ONLY $59^00

You can program any words or sentences • Adjust volume and pitch • Make adventure games that talk • Real sound action games • Make customized talkies • (Demo disk or tape included) • Requires Speaker

You can add **TEXT TO SPEECH SOFTWARE** that allows you to simply type what you want to hear!! Also allows you to add sound and voice to **SCOTT ADAMS AARDVARK** and **"ZORK" ADVENTURE GAMES** List $29.95 Sale $19.95 (Disk or Tape).

• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

COMMODORE 64

80 COLUMN BOARD

SALE

FOR ONLY $99^00



Now you can program 80 columns on the screen at one time! Converts your Commodore 64 to 80 columns when you plug in the PROTECTO 80 Expansion Board. List $199.00. **Sale $99.00.**

FOR ONLY $24.95 you can get an 80 Column Board "WORD PROCESSOR" with mail merge and terminal emulator PLUS! AN ELECTRONIC SPREAD SHEET (like Visicalc) List $59.00. **Sale $39.90.**
*If purchased with board only $24.95. (Tape or Disk.)

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

WE LOVE OUR CUSTOMERS

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Sale
$39.95
List $69.00

Program Your Commodore 64 In Plain English

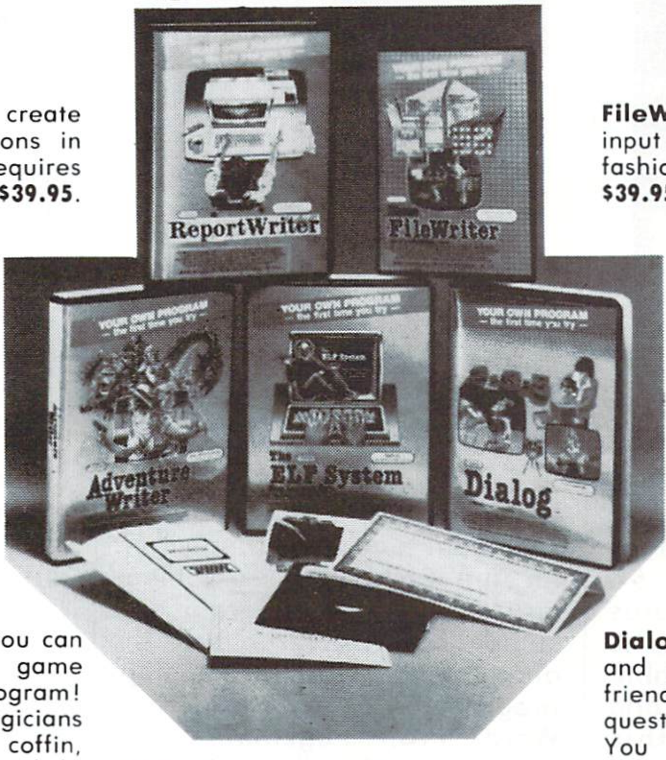
(No need to know Basic)

New Program Generators!!

CodeWriter

Information cann be a wonderful thing. Having enough of it when it's needed can save time and money and give you that very nice "together" feeling. **FileWriter** lets you control information so you can enter it, find it and change it — just the way you'd like. Whether it's information about bills, taxes, recipes, club membership, stamp collections, or your new home business, **FileWriter** puts things in order. Simply "draw" any kind of screen you want — just like making up a paper form to hold information. Only there's no paper to slow things down. Once you have your form, add some real computer power: calculate automatically, hold grand totals, test for bad information — for one form or hundreds. There's no need to "program". Just use plain English!! Most of the instructions you'll need are right on the screen for you. Like all CodeWriter programs, your **FileWriter** system gives you your own program on your own disk — in minutes the first time you try. Your new program will search for information any way you'd like, update old information, make new calculations, all at computer speed.

Program Generators!!



Report Writer — Lets you create output formats and calculations in any fashion you please. Requires FileWriter. List $59.00. **Sale $39.95**. (Disk).

FileWriter — Lets you create the input screens and calculations in any fashion you please. List $59.00. **Sale $39.95**. (Disk).

SALE 39.95

SALE 39.95

Adventure Writer — Now you can create your own Adventure game without knowing how to program! Create heroes, villains, magicians and monsters, castles and coffin, even dragons. All in plain English. When your creation is done Adventure Writer will write the program for you on your own disk. Fantastic Adventure tool! List $59.00. **Sale $39.95**. (Disk).

Elf — The visible worksheet is here! Now you can create applications screen by screen that calculates and prints out reports. Buy vs. lease, amortization, comparisons, and summations are just a few of the freeform spreadsheet functions you can do with Elf. List $59.00. **Sale $39.00**. (Disk).

Dialog — Now you can make quizzes and problems for your kids and friends. Make trivia games and other question and answer games for all. You pick the rewards and the punishments. Fantastic teaching aid for kids and party game maker for adults. List $59.00. **Sale $39.95**. (Disk).

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

The Most Practical Software For Your Commodore-64

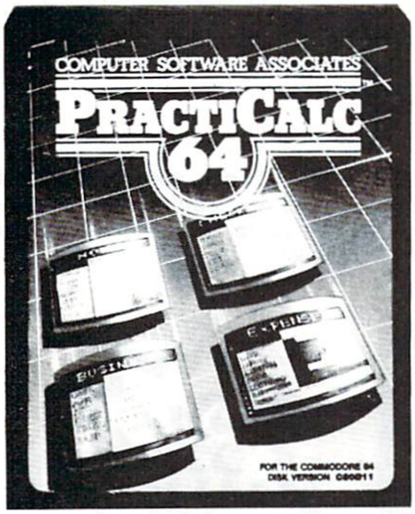




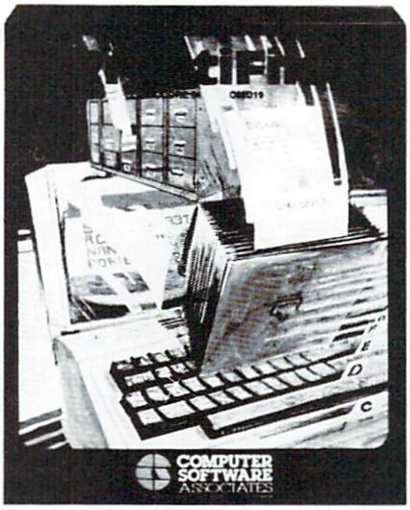
• Track Expenses, inventories, investments • Make Charts and Graphs • Project Profits

• Keep Mailing Lists • Change Records, numbers, methods of filing • Information Retrieval

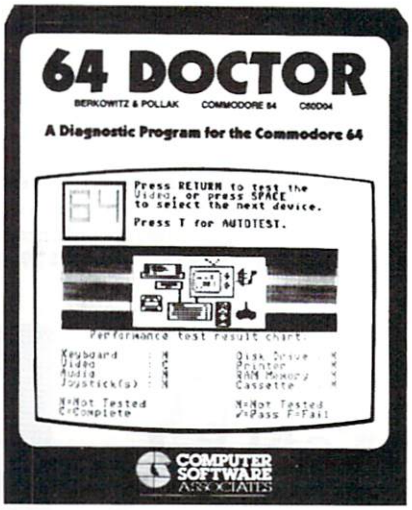
• Test RAM Memory • RS-232 Port • Keyboard • Video • Audio • Joystick • And More!



Practicalc 64: A consistent best seller, Practicalc 64 has become a reference standard among Commodore 64 spreadsheets. With features like alphabetic and numeric sorting and searching, variable comumn widths, graphing and 23 math functions, this program is an exceptional value. Practicalc 64 also interacts with Practifile, forming the perfect small business bundle. List $59.95. **Sale $46.95**. Disk. ($39.95 Tape.)



Practifile: Flexibility and large capacity make Practifile the ideal data-base manager for the Commodore 64. Files written with the program are compatible with Practicalc 64 and popular word processing programs such as EasyScript, Word Pro, PageMate and PaperClip. Finally, a full-featured data-base at an affordable price! List $59.95. **Sale $46.95**. Disk. ($39.95 Tape.)



64 Doctor: A special diagnostic program for the Commodore 64, 64 Doctor takes the guesswork out of isolating troublesome hardware problems. The program tests each piece of hardware to pinpoint defects and help prevent costly and time-consuming service calls. An essential program for all Commodore 64 users! List $34.95. **Sale $24.95**. ($19.95 Disk.)

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO ENTERPRIZES (WE LOVE OUR CUSTOMERS)

BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Reader Service No. 156

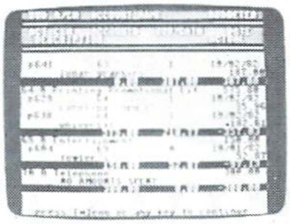
FUTUREHOUSE

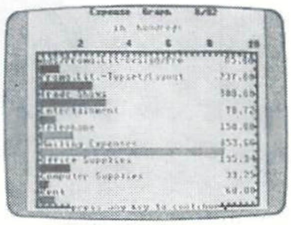
Introductory Offer!

Complete Personal Accountant
"Three Programs for the Price of One"
CHECKBOOK - BUDGET MANAGER - BILL PAYER

This program has become the most popular Commodore-64 program for money management in the home and in small businesses. Much more than a record keeper the complete personal accountant helps you plan, budget, monitor, and record your finances and your time. Chart of Accounts; Detail and Summary Budget Analysis; and Appointments/Payments Calendar are yours for one low price. List $79.00. **Sale $49.00.** Introductory Offer







Organize and monitor your finances with a flexible Chart of Accounts, and an option-packed Checkbook Maintenance program. Only the CPA offers on-screen editing for quick and easy corrections, check search capabilities in each field, and the capacity to print personalized checks.

With Detail and Summary Budget Analysis, and Net Worth Income Expense Statements, you can easily determine where your money is, where it's going, and where it's coming from. Finance 2 will automatically generate professional reports that are indispensable tools in making financial decisions.

With the Appointments and Payments Calendar you can schedule and then monitor your time and your money ... or graphically display your finances with the Color Graph program. Extra features let you organize names and addresses with our powerful Mailing List that features a global search capability.

Fantastic Educational Programs

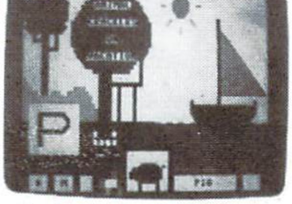
Little Red Riding Hood — TALKS!! — This famous bedtime story comes alive withgraphics, sound, text and SPEECH! Now you and your kids can relive the story of Little Red Riding Hood and improve your reading skills at the same time. Excellent for all ages!! List $34.95. **Sale $24.95** (Disk).





Peripheral Vision — At last a fantastic artists program for your light pen. Draw, erase and save in 14 colors. Zoom in and out, draw or fill circles, squares, triangles, or rectangles with the touch of a pen. Fantastic graphics tool. List $39.95. **Sale $27.95** (Requires light pen).

Animal Crackers — This program helps develop letter and word recognition skills by creating a screen full of animated objects, each representing a letter of the alphabet. Your child selects a letter with the joystick or light pen and suddenly an animal or object will appear with the correct spelling. List $34.95. **Sale $24.95** (Light pen or joystick required) (Disk).





Alphabet Construction Set — This is the most Advanced Alphabet Construction Set on record. The computer voice (it speaks) guides your child through every step as he/she draws the letter on the screen with a light pen. The computer analyzes the letters your child draws, points out mistakes and then suggests the necessary corrections! Fantastic Educational Value. List $34.95. **Sale $24.95**. (Light pen required) (Disk).

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Turn Your Commodore-64 Into A Sophisticated Musical Instrument

"The Program That Gives You A Reason To Buy A Commodore-64."

New York Times.

MusiCalc 1

Synthesizer & Sequencer

SALE 39.95

Synthesizer & Sequencer

This 1st step turn your Commodore-64 into a sophisticated musical instrument — a three voice synthesizer and fully interactive step sequencer play along with pre-recorded songs or develop your own and record the music you create.

List $59.95. **Sale $39.95.**

With Musicalc **anyone** can • Make and record sophisticated music • Print out sheet music from your creations • Turn your computer into a keyboard • **No experience necessary!**

MusiCalc 2

ScoreWriter

ScoreWriter

Combine with Musicalc 1 and a graphics printer (Super-10) to produce sheet music from your original composition. List $39.95. **Sale $29.95**. Coupon $24.95.

MusiCalc 3

Keyboard Maker

SALE 39.95

Keyboard Maker

Turns your Commodore-64 into a musical keyboard. Comes with over 30 pre-set keyboard scales from Classical to Rock. Requires Musicalc 1. List $39.95. **Sale $29.95**. Coupon $24.95.

You will sound great with Musicalc. To prove it we will send you **FREE** a **record** with sounds created with a Commodore-64 and Musicalc by people without instrument or musical experience. To get your Free Record write or **call 312/382-5244**.

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

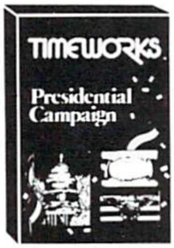
PROTECTO ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Free Program Offer* TIMEWORKS

Home Management Educational Software

Presidential Campaign

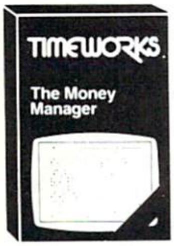
Start with $2,500,000 and a Presidential Nomination. Spend your money wisely to conduct a nationwide campaign to win the confidence of the American Public. At the end of the campaign the American Public will decide on the final decision. Fantastically Realistic. List $29.95. **Sale $19.95**.



**** Buy the Fantastic Presidential Campaign and choose any other Timeworks Program on this page FREE!!***

Tape or Disk

Expires 10/1/84



Money Manager

16 categories can be input on a monthly basis for a 12 month period. Full analysis including budget and actual comparisons plus detail by individual items. Colorful bar charts show where you're spending to much or too little. Record all information to tape or disk. List $24.95. **Sale $19.95**.

The Electronic Checkbook

Check recording, sorting and balancing system. Organize and record checkbook transactions for easy recall and sorting by category. A must for tax time. Also produce the sum and average for transaction amounts by indexed categories. Interface with the MONEY MANAGER for graphic detail and actual-budget comparison. List $24.95. **Sale** $19.95.



Your Choice $19.95

Programming Kit I

Learn Basic Programming from the ground up. This first step takes you through the actual design of Lunar Lander, using moving graphics.



Shows in detail how basic programming works and what should go into a good game. List $24.95. **Sale $19.95.**

Your Choice $19.95

Programming Kits

Manuals are written in plain, easy to follow English. The 8 step approach to program design is fully explained and can easily be applied to your future programming adventures.

Programming Kit II

Intermediate game design and Sprite Builder. Design Slot Machine and study intermediate programming including arrays, subroutines, sound, special function keys, and random function. Plus Sprites are covered in detail and a Sprite Editor is included. List $24.95. **Sale $19.95.**

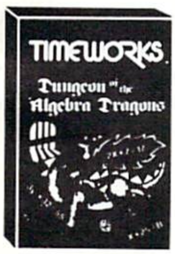


Programming Kit III



Intermediate Data Base design. Instead of games you may want to go the business route. Use this program to develop a Data Base Manager to allow quick storage and retrieval of any information. Learn disk/tape storage, sorting, menus, plus multi dimensioned arrays, plus a whole lot more. List $24.95. **Sale $19.95.**

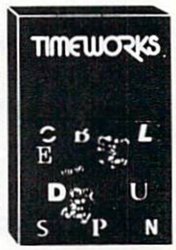
Dungeons Of The Algebra Dragons



At last a good educational game. You must wander through 3-D mazes until you encounter dragons, then you must answer the algebra problems they give you. A right answer gets you a free ride, a wrong answer gets you eaten. Fantastic educational value. (Tape or Disk.) List $29.95. **Sale $24.95**.

Spellbound

The screen is sprinkled with the letters from one of 20 words of your choice along with a few extra random ones. Use your Joystick to sting the letters in the right order but don't sting the wrong ones or cross your path or you explode. Fantastic educational value. Fun for all ages. List $24.95. **Sale $17.95.**



Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery. 2 to 7 days for phone orders. 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO

PROTECTO ENTERPRIZES (WE LOVE OUR CUSTOMERS)

BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Reader Service No. 161

Commodore 64

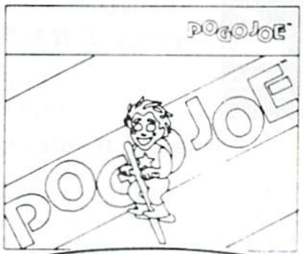
screenplay

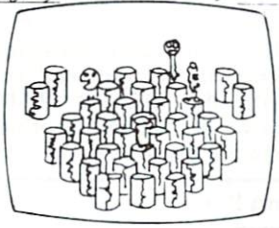
Superior Computer Games

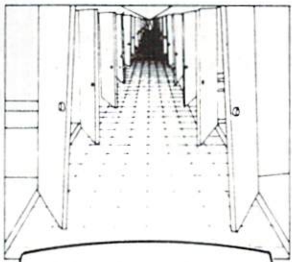
Pogo Joe

Better than Qbert. Guide Pogo Joe across 32 screens of cylinders. As many as 8 nasties at a time are out to kill you. Fantastic graphics and sound. List $29.95. **Sale $19.95**. Coupon $16.95. (Tape or Disk)

• GAME OF THE YEAR!









Asylum

One of the best graphic adventures we've seen. Get yourself out of the Asylum past guards and secret doors. Stunning, 3D graphics will make you feel like an inmate. List $34.95. **Sale $24.95**. Coupon $22.00. (Tape or Disk)



Dunzhin

A good sword, little magic, and lots of cunning takes you against Evil Wyverns, Saber Toothed Tigers, and the Undead. If you reach the bottom of the dungeon your halfway home. Great and 30 graphics. List $34.95. **Sale $24.95**. Coupon $22.00. (Tape or Disk)

Wylde

Be a Hobbit! Make your way through the magic mountains to the treasure of the mountain king. Fight Trolls, Orks and other nasties along the way. Fantastic realism. List $34.95. **Sale $24.95**. Coupon $22.00. (Disk)



Playful Professor

Make your way to the treasure by correctly answering math problems. If you guess wrong and the professor shows you how step by step. Addition, substraction, multiplication, and division basics are all covered. Fantastic educational value. List $34.95. **Sale $24.95**. Coupon $22.00. (Tape or Disk)





Ken Uston's Professional Blackjack

The ultimate Blackjack tutor made famous by Ken Uston, $5,000,000 winner. All different casino Blackjacks are discussed as well as a free $29.95 value Blackjack Book are yours. Fantastic tutor. List $79.95. **Sale $49.95**. (Disk)

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

PROTECTO

ENTERPRIZES (WE LOVE OUR CUSTOMERS)

BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Reader Service No. 162

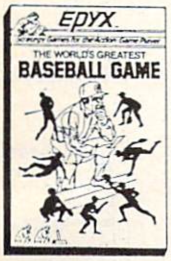
EPYX
COMPUTER SOFTWARE
Strategy Games for the Action-Game Player

*Buy 1 (at Sale Price!) Get CRUSH CRUMBLE CHOMP FREE

Commodore 64

(Expires 10/1/84)

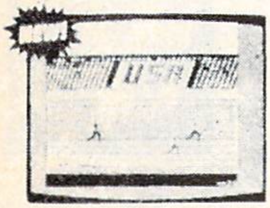
Award Winning Games



World's Greatest Baseball — Finally a great sports game. Realistic right down to the playing field. Control all the actions--even the outfielders. Play realistic baseball all year round. (Fantastic graphics, action, and sound!) (Disk) List $34.95 **Sale $24.95.**

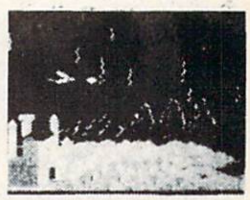
Fax — Stop wasting quarters! Now you can play Exidy's Fax at home. Play against the computer or an opponent to get the answer first, over 3,700 questions dealing with sports, television, movies, histories, and many more subjects will keep you playing for hours (a must for trivia buffs). List $34.95. **Sale $24.95**. (disk only).

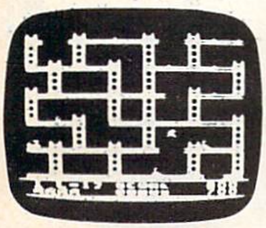




Summer Games — Now you can compete in the Olympics right along with the real competitors. Compete in track, swimming, diving, cycling, archery, weight lifting, rowing and more. This Olympic game is so realistic it even includes a beautiful opening ceremony and an awards presentation after each event. (Highly recommended, by far the best Olympics game we've seen). List $39.95. **Sale $28.00**. (disk only).

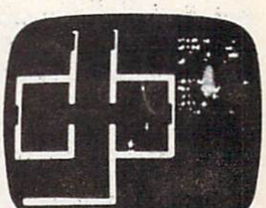
Dragon Riders of Pern — Based on the best selling book series by Anne McCaffey (over 10 million sold). This program forces you to make an alliance with the Lord holders, then mount your dragon (stunning graphics) and battle the dreaded thread. (Super fast action with a great theme). List $39.95. **Sale $24.00**. (tape/disk).





Jumpman — If you like Donkey Kong, you'll love Jumpman. Over 30 different screens with 8 speeds and 5 skill levels make this the fastest action game in the country. You must leap girders, climb ropes, and scale ladders to reach and diffuse bombs while avoiding robots, birds, bullets, explosives, crumbling walls, vanishing escape routes, and many other obstacles. (Truly a fantastic game!) List $39.95. **Sale $24.95** (Disk/Tape.)

Temple of Apshai — ***(computer game of the year)***. This is the standard by which other adventure games are judged. Full color graphics portray the temple and all its contents — magic, monsters, doomed cities and damsels in distress. Do battle in real time with over 20 monsters, expansion modules will keep your adventure alive in the future. List $39.95. **Sale $24.95**. (tape/disk).



(Also Available For VIC-20)

Add $3.00 for postage. Add $6.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. We accept Visa and Master-Card. We ship C.O.D.

PROTECTO
ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Reader Service No. 163

SUPER
Fantastic Sale!
HES Special Programs For VIC-20 Computer

While They Last!! Call Today!! First Come Basis!!

SYNTHESOUND Sale $9.95
• Fantastic Sound & Music • Piano Keyboard

Now your VIC-20 can be used as a state-of-the-art music synthesizer and special effects generator! A very user friendly manual guides you through sessions of learning and development of your synthesizer "style"! You can program Synthesound to sound like many different instruments like bagpipes, organ, bass and many others. Synthesound brings technical musical abilities to any VIC-20 user. Prepare to step into the exciting world of the futuristic computer musician.

- VIC keyboard used as a 2 octave, fast attack keyboard.
- 3 voices with 4½ octave range.
- 4 low frequency ocillators.
- 4 envelope shaping controls (Attack, Decay, Sustain, Release).
- 4 voice/20 note sequencer for repeating melodies or rhythms.
- High resolution piano keyboard display.



List $39.95
Sale $9.95

SHAMUS (No. 1 Joy Stick Action Game!) Sale $9.95

Fight against the Roboroids and Snap Jumpers as you wind your way through the Shadows lair. Dangerous obstacles may mean your doom if you're not fast with your joy stick action! Excellent high resolution graphics and fantastic sound! VERY ADDICTIVE!

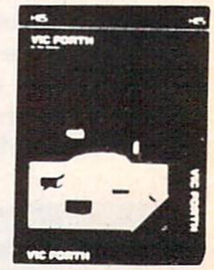


List $39.95
Sale $9.95

VIC FORTH LANGUAGE Sale $19.95
Faster than BASIC — easier than Assembly language — Language of Robots.

VIC FORTH is an exciting new cartridge for your VIC-20 computer. You now have a language that is more powerful than BASIC and easier to program than assembler! VIC FORTH is a nearly complete implementation of the "FORTH INTEREST GROUP" (fig.) version of FORTH. The VIC FORTH editor follows closely the standard FORTH editor described in the book, "Starting FORTH", plus it has many, many additional features.

- Full screen editor which has 16 lines of 64 characters each (standard FORTH Screen) through use of a horizontal scrolling window.
- User definable words; this means a function not already supported can be created by you and added to VIC FORTH for future use.
- Full VIC sound and color capabilities.
- Runs in standard 5k VIC.
- Output to printers and disks and tape machines.
- Up to 24k additional memory can be added.
- 31 characters for variable names.

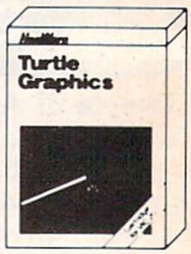


List $59.95
Sale $19.95

TURTLE GRAPHICS Sale $19.95
• Create colorful pictures and graphics • Learn how computers compute.

Turtle Graphics is an exciting and fun way to discover computer programming concepts. Turtle Graphics is an easy-to-learn computer language. It enables the beginning programmer to master the computer in just a few minutes. By telling an imaginary turtle how to roam about on the VIC's screen, the Turtle Graphics programmer can create colorful and artistic pictures. As one becomes more and more adept at teaching the turtle to paint using the VIC's rich graphics and color palette he or she is painlessly learning all of the basics of computer programming. This not only develops computer awareness, but enriches one's math, logic, communication and artistic skills as well.

- Turtle Graphics has over 30 different commands.
- Programs may be listed on a printer and saved on or loaded from tape or disk.
- Turtle Graphics is menu driven for ease of use.
- Trace mode to help the beginning programmer follow the logic of his program one step at a time.
- Well written, easy-to-follow manual with many examples included.



List $49.95
Sale $19.95

We Have All HES Products At Super Sale Prices!!

PROTECTO ENTERPRIZES (WE LOVE OUR CUSTOMERS)
BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

Reader Service No. 164

VIDEO RAM

LOOKING AT THE SCREEN'S MEMORY

BY DALE RUPERT

This month we will examine the video display and its associated memory locations. If your time is limited, take this quick quiz to determine whether you need to read any further into this article:

1. How are the following statements similar?

```
A) POKE 1024,1
B) PRINT CHR$(19)CHR$(65)
C) POKE 53272,23 : POKE 1024,65
```

2. If PEEK(55296)=2, what color is the character that is in the upper left corner of the screen?

3. Which statement would put a reverse-video "9" on the screen?

```
A) PRINT "{RV} 9"
B) POKE 2000,57+128
C) PRINT CHR$(18)CHR$(57)
```

Now check your answers carefully. All three statements in question 1 will put the letter "A" in the upper left corner of the screen, sometimes. B) always works. A) and C) work depending upon the value in the corresponding color memory location 55296. (Be generous with the partial credit.)

If the statement in question 2 is correct, you can be sure that the character there is red, if there is one there. On the other hand, the character may be red and PEEK(55296) may have a completely different value. Confusing, isn't it?

The answer to question 3 is that A) and C) will always display a reverse-video "9", and B) will sometimes, depending upon the values in locations 53272 and 56272.

If you got all problems correct, congratulations, and so long. See you next month.

If all this stuff is confusing beyond comprehension, welcome! After studying this article, you will be able not only to understand the video and color memory, but to manipulate it as well. It's really not nearly as incomprehensible as the answers to the little quiz implied.

SCREEN COORDINATES

To begin, let's look at the way the video display is organized. Although we'll be dealing with the Commodore 64, most of the concepts (but not the numbers) also apply to the VIC 20. It will be helpful if you get your *C-64 User's Guide* and turn to Appendix G or your *C-64 Programmer's Reference Guide* and turn to Appendix D. The screen memory map also shows a 40 by 25 matrix of cells. The cells are numbered sequentially 1024 through 2023 from upper left to lower right.

There is a memory location at address 1024 in RAM which stores one 8-bit byte. Whatever is in that memory cell is interpreted by the computer to be the code for a character which is to be displayed on the screen. For example, if there is a 1 in address 1024, there will also be a letter "A" in the upper left hand corner of your screen.

Want to see for yourself? Press <SHIFT> and <CLR/HOME> simultaneously to clear the screen. Move the cursor down a few lines with the <CRSR DN> key. Then type POKE 1024,1 and press <RETURN>. What you have done is put a number 1 into the memory at address 1024. Do you see the "A" in the upper left corner? You don't? Try moving the cursor up to the corner with the <SHIFT> and <CRSR UP> keys. Now you should see an "A" flashing with the cursor.

What happened is that we put an "A" up there all right, but it had the same color as the background so it was invisible. The cursor changed the character to its reverse-video image and made it momentarily

visible. So we now know how to put invisible characters on the screen. Could you put an "A" in the lower right-hand corner? POKE 2023,1 should work. Be careful when you move the cursor down to look at it that the screen doesn't scroll. If it does, you'll find an invisible "A" somewhere above that corner.

Notice on your screen memory map that the start of each row on the screen has a memory value which is 40 greater than the value of the row above it. The last cell in the top row of the screen has an address of 1063, and address 1064 is the first cell in the second row on the screen. The cell directly below 1063 is (1063 + 40) or 1103.

We can't do very much with invisible characters, so let's make them visible. Associated with each screen memory cell is a corresponding color memory cell. Look at the color memory map in your book. The grid is identical in shape and size to the one for screen memory. The only difference between the two is the numbering. You may put a number from 0 to 15 into the color memory cells which corresponds to the color of the character which will appear in each cell.

Press <RUN/STOP> and <RESTORE> together to produce the default colors on your display. Clear your screen and move the cursor down a few rows. Enter the statement

```
POKE 55296,0
```

Did the upper left corner of your screen turn black? No? Try also typing

```
POKE 1024,1
```

Now you should see a black "A" in the upper left corner.

The color memory and the screen memory work together. The character in each screen location is displayed in the color as specified by the corresponding color memory value. If you now type

```
POKE 55296,1
```

you should see a white "A". The values of the colors are listed above the color memory map in your reference book.

To put a red "A" in the lower left corner of the screen, type

```
POKE 1984,1 : POKE 56256,2
```

About the only thing we haven't covered is how to determine the value to POKE if we want something besides the letter "A". That's where the *Screen Display Codes* appendix in your reference book comes in handy (Appendix E in the *User's Guide* and Appendix B in the *Programmer's Reference Guide.*)

There are two sets of screen codes and their POKE values listed in that table, Set 1 and Set 2. When you turn your computer on, Set 1 characters are automatically selected. If you press the <SHIFT> and <COMMODORE> keys simultaneously, you can toggle over to Set 2 or back again. All the letters on your screen change to lower case when Set 2 is selected.

You should be able to see from this table where the value of 1 for the letter "A" came from. If you had POKEd a 2 into address 1024, you would have seen a "B". POKE a 94 into address 1024 (POKE 1024,94) and you should see the pi symbol. Now press the <SHIFT> and <COMMODORE> keys and the pi becomes a checkerboard pattern. You will be able to find both symbols in the Screen Codes table of your book.

It is easy enough to change from one character set to another from the keyboard, but how about from within a program? That is also very easy. If you want to select Set 1, simply use the statement POKE 53272,21 and to switch to Set 2, use POKE 53272,23. With the pi in the upper corner of the screen, type POKE 53272,23 and see that it changes.

Now you should be able to understand parts A) and C) of question 1 in the quiz. Not really that difficult after all! In C) we simply selected Set 2 and put character 65 (which is an "A") into location 1024. Question 2 should be reasonably obvious also. Since location 55296 is the color memory location for the upper left corner of the screen, and since 2 is the code for the color red, a character in that position will be red.

Answer 3 C) to the quiz should look somewhat familiar. Address 2000 is in the screen memory map on the bottom row. A note at the end of the Screen Display Codes appendix indicates that reverse-video characters have a value which is 128 greater than the normal-video characters. Consequently POKE 2000,57+128 puts a reverse-video version of character number 57 (a "9") on the bottom row of the screen, assuming Set 1 is selected.

SOME ACTION

Enough talk. How about some action! Let's put a whole row of characters on the screen. Try this:

```
5 PRINT CHR$(147)
```

```
10 UL=1024 : UR=1063
20 FOR SM=UL TO UR
30 POKE SM,1
40 POKE SM+54272,0
50 NEXT
```

Line 5 clears the screen. Line 10 defines the Upper Left and Upper Right screen memory addresses. The FOR-NEXT loop in lines 20-50 POKEs character 1 into all locations from UL to UR. The color memory is always 54272 greater than the corresponding screen memory, so the 0 in line 40 causes all characters to be black on the screen.

Add lines 15 and 60 through 90 as shown below:

```
5 PRINT CHR$(147)
10 UL=1024 : UR=1063
15 LL=1984 : LR=2023
20 FOR SM=UL TO UR
30 POKE SM,1
40 POKE SM+54272,0
50 NEXT
60 FOR SM=UR TO LR STEP 40
70 POKE SM,1
80 POKE SM+54272,0
90 NEXT
```

Line 15 defines the other two corners of the screen. The new FOR-NEXT loop puts A's down the right side of the screen. Notice the STEP size of 40 in line 60. To go from right to left along the bottom of the screen, we must use a STEP of -1. To draw a vertical line from the bottom of the screen to the top, use a STEP of -40. To check your understanding of what we've covered, try adding some statements to this program to draw a box completely around the screen.

A few modifications, as shown in the *Spiral* program below, takes these concepts one step further, and create an interesting graphic display.

```
5 PRINT CHR$(147)
10 UL=1024 : UR=1063
15 LL=1984 : LR=2023
20 C=RND(0)*256
25 FOR SM=UL TO UR
30 POKE SM,C
40 POKE SM+54272,11
50 NEXT
60 LR=LR-41 : IF LR<1024 THEN10
70 FOR SM=UR TO LR STEP 40
80 POKE SM,C
90 POKE SM+54272,11
100 NEXT
110 LL=LL-39 : IF LL<1023 THEN 10
120 FOR SM=LR+1 TO LL STEP -1
130 POKE SM,C
140 POKE SM+54272,11
150 NEXT
160 UL=UL+41 : IF UL>2023 THEN 10
170 FOR SM=LL TO UL STEP -40
180 POKE SM,C
190 POKE SM+54272,11
200 NEXT
210 UR=UR+39 : IF UR>2023 THEN 10
220 GOTO 20
```

Line 20 picks a random character between 0 and 255 inclusively. The IF statements in lines 60, 110, 160, and 210 are "safety valves" which make sure that the memory addresses which will be POKEd do not exceed the screen boundaries. Poking data into the wrong places can definitely "lock up" your computer. In the worst case, you have to turn the power off and then on again. Those IF's guarantee that such a thing never happens in this program. As always, be sure to SAVE your program *before* you run it.

You should be able to understand the concepts used by this program. The "+1" in line 120 is there simply for aesthetic reasons. You can easily replace the 11's with a variable so that the color changes as well as the character. Use a line similar to line 20 to randomly pick a number between 0 and 15 for the color memory value.

There are many other ways you can modify this program with very interesting results. Changing step sizes and rearranging the lines similar to line 60 which vary the locations of the corners will drastically change the results. Whatever you do, leave the IF statements!

With the POKE statement, you put a number directly into a specific memory location. If that memory location happens to belong to the screen memory, the computer does the rest of the work to display the results on the screen. the PEEK statement allows you to read the value stored in a particular memory location. If you PEEK at an address associated with video memory, you will get a code number telling you what is on the screen at the corresponding location. Who said PEEKs and POKEs were complicated?

GETTING COORDINATED

Quick—what is the address of the fourteenth column of the twelfth row on the screen? Yes, it is

somewhere between 1024 and 2023, but where exactly? Even with the memory map in front of you, it is a bit tedious to figure out.

So far the most difficult thing we've had to do is look up the memory address of a particular location on the screen. Let's make it even easier. We can picture the video screen as a Cartesian coordinate system—you remember Descartes with his X's and Y's, with slopes and intercepts, don't you? If not, it doesn't matter. The computer will do all the work anyway.

We will use the COLUMN numbers and the ROW numbers shown on the memory maps in your reference book. Column numbers run from 0 to 39, left to right, and Row numbers run from 0 to 24, top to bottom. In other words, the origin (location 0,0) is in the upper left corner of the screen. That's different from the standard Cartesian coordinate system, but we can let the computer take care of that also.

A fairly straightforward formula converts a COLUMN, ROW pair of numbers into the corresponding Screen Memory address. The formula is:

```
SM = 1024 + COLUMN + 40 * ROW
```

or if you prefer to think in terms of X and Y, where X is the horizontal distance from the left, and Y is the vertical distance from the top:

```
SM = 1024 + X + 40 * Y
```

For example, the fourteenth column and the twelfth row has a memory address of 1518 as you can quickly see if you type

```
PRINT 1024 + 14 + 40 * 12
```

in direct mode.

You could create a simple subroutine to plot a point anywhere on the screen. Of course this is low resolution, but as you saw from the *Spiral* program above, it can still be impressive. Your main program would specify an X and Y value. Then call a subroutine similar to this to plot each point:

```
1000 SM = 1024 + X + 40 * Y
1010 POKE SM, 81
1020 POKE SM + 54272, 1
1030 RETURN
```

The 81 corresponds to a circle in Set 1, although you could use any symbol you want. You could even use a variable if desired. Similarly the color value in line 1020 could be anything from 0 to 15.

As mentioned earlier, we can let the computer change the coordinates around so that 0,0 is in the lower left corner of the screen, just as the math books show. Simply change the Y in line 1000 above to (24 − Y). It looks like this:

```
1000 SM = 1024 + X + 40 * (24 - Y
)
```

Now when Y equals 0, the computer uses 24, and when Y equals 24, the computer uses 0. All the other values are properly changed as well.

The following program uses these concepts to draw a line between any two points on the screen. The program uses the Bresenham Algorithm which is described in a book by Newman and Sproull called *Principles of Interactive Computer Graphics* (Second Edition, McGraw-Hill, 1979). The algorithm generates a uniformly distributed set of points along the line from one point to another. The BASIC adaptation here won't win any speed records, and once again, we are only dealing with low resolution character graphics, but it is a reasonably efficient, general purpose, line-drawing program.

```
5 :REM LINE DRAWING ROUTINE
6 :REM FOR LOW RESOLUTION GRAPHIC
S
7 :REM USING THE BRESENHAM ALGORI
THM
8 :
10 PRINT CHR$(147)
11 :REM SPECIFY X1,Y1 AND X2,Y2 H
ERE
12 :
13 :REM  X=0 TO 39 : Y=0 TO 24
14 :
18 :REM   FOR EXAMPLE...
19 :
20 X1=0 : Y1=0 : X2=39 : Y2=24
30 GOSUB 100
40 X1=39 : Y1=0 : X2=0 : Y2=24
50 GOSUB 100
55 :REM   DIAMOND
60 X1=10 : Y1=12
70 READ X2,Y2 : IF X2=-1 THEN 99
80 GOSUB 100 : X1=X2 : Y1=Y2 : GO
TO 70
90 DATA 20,6,30,12,20,18,10,12,-1
,-1
95 :
99 END:=====================
```

```
100 DX=X2-X1 : DY=Y2-Y1
110 IF ABS(DX)>ABS(DY) THEN D2=DX
 : D1=DY : C2=X1 : C1=Y1 : N=1 :G
OTO 130
120 D2=DY : D1=DX : C2=Y1 : C1=X1
 : N=0
130 S2=SGN(D2) : S1=SGN(D1) : S3=
S1*S2*D1*2 : E=S3-D2 : A2=ABS(D2)
 : D4=2*D2
140 FOR J=1 TO A2
150 IF N=1 THEN SM=1024+C2+40*C1:
GOTO 170
160 SM=1024+C1+40*C2
170 IF SM < 1024 OR SM > 2023 THE
N 210
180 POKE SM,102 : POKE SM+54272,1
190 IF E*S2 > 0 THEN E=E-D4 : C1=
C1+S1
200 E=E+S3 : C2=C2+S2
210 NEXT J
220 RETURN
```

It is possible to speed the program up somewhat by removing spaces and compressing statement lines. The only lines which are worth compressing are in the FOR-NEXT loop starting at line 140. That's where most of the action is. Be careful of the IF-THEN statements if you combine any lines.

To use this program with the "normal" coordinate configuration (0,0 in the lower left corner of the screen), replace C1 in line 150 with (24 −C1), and replace C2 in line 160 with (24 −C2) just as we did in line 1000 above. Change the plotting character and its color in line 180, if desired.

CHARACTER PRINTING

So far, we have dealt with POKE graphics. This is a random-access type of graphics, analogous to a floppy disk. Each screen location is as accessible as any other. PRINT graphics, on the other hand, are more sequential in nature, analogous to cassette storage. To get to a particular spot on the screen, we must move the cursor through sequential locations. If you are putting blocks of characters on the screen, such as words, then character printing is easier and preferable to POKEing.

Once the cursor is in position, there is less to worry about with printing characters than with POKEing them, since the computer keeps track of updating the cursor. With POKE graphics, the cursor is not even affected, and it is up to the programmer to keep track of where the next character will appear.

The Commodore computers allow strings composed of graphics characters to be contained in programs and easily put onto the screen. For example, if we type:

```
PRINT "<SHIFT-A><SHIFT-S><SHIFT-Z
><SHIFT-X>"
```

(where <shift-A> means hold down the shift key and press the A key), the four card suits are printed on the next screen line. To put them in the lower right corner of the screen, we must first position the cursor and then execute the PRINT statement.

The cursor movements can also be contained within the quotation marks or they can be assigned to string variables. We might have a statement such as U$="<cursor up>" where we pressed the shift key with the cursor up/down key inside the quotation marks. Now whenever we execute PRINT U$, the cursor will be moved up a line.

The disadvantage of putting the graphics and cursor controls within quotation marks is that they are difficult to interpret and debug later. An alternative is to use variables as above, and also to use the CHR$ function. To move the cursor up one line and print a heart symbol, we could use this statement:

```
PRINT CHR$(145);CHR$(115)
```

rather than PRINT " <cursor up> <shift-S> ". They do the same thing, although the first statement is easier to interpret in a listing. The disadvantage of the first statement is that you must look in a book to see what CHR$(145) is.

If you look in the *ASCII and CHR$ Codes* appendix in your reference book, you will see a list of keys or symbols and their corresponding ASCII (pronounced ASK-EE) values. CHR$(115) represents a heart shape and CHR$(145) represents a cursor-up command.

Now you should be able to figure out the remaining answers to the quiz at the start of this article. Answer 1 B) moves the cursor home with CHR$(19) and prints an "A" with CHR$(65). If CHR$(147) had been used instead of CHR$(19), the Clear/Screen function would have been performed. Note that the shifted keystroke has an ASCII value which is 128 greater than the unshifted keystroke.

Answers 3 A) and 3 C) are clearly identical. The <RVS ON> function is duplicated by printing CHR$(18), and CHR$(57) corresponds to the "9" digit. Note that the numerical digits have ASCII values which are 48 greater than their numeric value.

Now you should be able to go back and pass the Low Resolution Graphics Quiz with flying colors. Each character cell that we have dealt with this month consists of much smaller dots, or pixels, on the video screen. Each character is actually an 8 by 8 block of pixels, some of which are on (or one color) while the others are off (or another color). Since there are 1000 character locations on the screen, and each location has 64 pixels, there are 64,000 dots which the computer can manipulate.

Your Commodore computer gives you the ability to specify precisely which of the 64,000 pixels are on and which are off. You can also specify their colors. But we will discuss just exactly how that is done some other month. In other words, that is left as a problem for the student. □

Sound Explorer

Continued from page 29

information you will ever see is presented to you on the screen. In order to modify any of the parameters in the chart on top of the screen, simply move the cursor over the appropriate parameter using the cursor control keys; then, enter the new value using the numbered keys and strike RETURN.

In order to vary the other functions, simply strike the function key indicated to the right of the current value. You will be prompted at that point to enter the new value.

Once you have created a sound that you wish to hear, press the spacebar. This starts the ADSR cycle. The volume of the oscillators will rise and fall as specified by attack and decay. At this point the sound will remain at the sustain level. When the spacebar is struck again, the envelopes will start their decay cycles.

One general note: The 64 has fantastic sound capabilities. In order to take advantage of them I very strongly recommend that you obtain an adapter to enable you to hook your 64 up to your stereo. You'll be amazed by the difference! □

SEE PROGRAM LISTING ON PAGE 90

Cassette Interface

Continued from page 34

sette earphone jack and convert it into a signal that the computer can use, that is, a five volt signal. The signal from the cassette recorder can vary anywhere from six volts to twelve volts maximum depending on the unit used. The best way to prevent loading problems is to keep the volume control at one level all the time. If you can keep the control at maximum for any brand of recorder, those loading problems that have driven people to disk drives can nearly be eliminated.

From the schematic we see that there is a resistor, two diodes, and a capacitor that the earphone signal must pass through to get to the two inverters. The signal from the earphone is AC, that is, it travels above and below ground level reference. With the volume control set at maximum, this signal is nearly a square wave at the amplitude of the recorder's voltage. When this signal meets those two diodes, it is prevented from going much below ground level, actually .6 volts, and it can't go above 5 volts. This is about what the inverter wants, so the D-4 Cassette Read pin now has a nice five volt signal that only varies in frequency.

The next thing we have to do is build the interface. As you can see from the photos, the circuitry is built into a small 2x4x1 inch box. The biggest problem I had in putting this device together was the connector that goes to the Commodore. This is a six-pin connector and cannot be found in any local electronics supply store. What I did was to sacrifice a 12 pin connector I had, and cut it down to the right size. Other sizes can also be modified (see parts list). This was then soldered onto a prototyping board which was cut to fit into the box. The six pins on one side have to be connected to the same six pins on the other side of the connector. It is a good idea to install a small piece of plastic in between pins B-2 and C-3 of the connector to key it so that it won't be connected the wrong way.

All the other components are mounted at various locations on the board and connected up. The box has two cutouts, one of either end to accept the cable and the part of the board that connects to the connector. The switch is also mounted in the box. If you cut the board just right, as well as the cutouts in the box, the board will fit very snuggly and no mounting hardware will be required. See the photos.

If you are not familiar with soldering and electronics, it is best to find a friend who is. You can try to follow my own layout from the photos, but this may be difficult. You have to use a low wattage soldering iron, and thin wire for all the interconnections. Since this interface is not a beginner's project, although it is simple, I cannot give detailed instructions on how to put it all together. As a last resort, you can purchase interfaces for cassettes from various sources, and I have seen them from $28 to $35.

The parts list show that you can purchase all of the parts from two sources. Since Jameco Electron-

ics has a minimum order of $10, you may want to buy everything you can from them, and everything else from Radio Shack. But I have not used the Jameco parts and cannot say what the sizes of some of their components are. However, there should be enough room in the box for everything.

I hope that the mysterious cassette port has been demystified. My next project is to connect my TRS-80 up to the Commodore. I hope to bring you those sessions since my first attempt is to use and demystify the RS-232 port. Then I will be able to not only print out listings through my TRS-80, but also to save programs on my disk drives. Until then, happy computering. □

COMMODARES

Continued from page 49

```
170 IF B>95 AND B<128 THEN E$=CHR
$(B+64)
180 PRINT#4,E$;
190 NEXT
200 PRINT#4,BS$
210 D=D+40:IF D>1984 THEN 230
220 GOTO 110
230 CLOSE 4:END
```

One final program this month comes from Larry Masterson (Willard, OH). This is not exactly a utility. In fact it's a bit difficult to classify at all. You PEEK and POKE experts should have a good time figuring out what makes it work anyway. What does it do? Well, you probably won't believe your eyes. You'll have to get out a mirror and stand on your head to read your screen once you run this program. After you are tired of standing on your head, either RUN/STOP-RESTORE or POKE 53272,21 will make things normal again. Very interesting, Larry.

```
2 :REM TURN YOUR SCREEN UPSIDE DO
WN!
3 :REM SUBMITTED BY LARRY MASTERS
ON
4 :
5 PRINTCHR$(142):POKE52,48:POKE56
,48
10 CLR:K=0:DIMA(511):X=7:Y=0
20 POKE56334,PEEK(56334)AND254
30 POKE1,PEEK(1)AND251
40 FORI=0TO511:A(I)=PEEK(I+53248)
:NEXT
50 FORI=XTOYSTEP-1:POKE12288+K,A(
I):K=K+1:NEXT:X=X+8:Y=Y+8:IFX>=51
2THEN100
60 GOTO 50
100 POKE1,PEEK(1)OR4
110 POKE56334,PEEK(56334)OR1
120 POKE53272,(PEEK(53272)AND240)
+12
130 INPUT"TYPE";A$
140 :::REM  POKE 53272,21  TO RET
URN TO NORMAL:::
```

In addition to the people mentioned above, we have received solutions to March and April *Commodares* from Alan Bowen (Franklin, TN), Ian Fraser (100 Mile House, BC), Elizabeth Romig (Melrose, MA), James Viscosi age 14 (Oriskany, NY), John Oyzon age 13 (Rome, NY), Jim Bernard (Springfield, VA), Dave Haxton (Mooresville, IN), Jean-Francois Conture (Chicoutimi, Quebec), Robert Bostwick (No. Platte, NE), HaRRob Van (Huber Heights, OH), Vince Cino (Silver Creek, NY), and Alan Scott (Morgantown, WV).

Don't let the lag between when your letters are received and when they're published discourage you! If you write every month, sooner or later you'll start seeing your name every month. □

Base Conversions
Base Conversions
Base Conversions

FOR THE C-64 AND VIC 20

By Drexel B. Gibson

All beginning programmers, when they reach a certain point in their training, need to learn to convert numbers from one base to another. I'd go so far as to say that assembly language programming is impossible without the ability to convert numbers between binary, decimal, and hexadecimal.

We are all familiar with the base ten, or decimal, number system. A three-digit number like 123 stands for 3+(2X10)+(1X10X10), with 10 being the base for the number system. It's the same in base two, called binary. 010 means 0+(1X2)+(0X2X2) or 2. Notice that there are only two digits used, 0 and 1. In base 16, usually called hexadecimal, we need 16 different digits, so we use the letters A-F to stand for 10-15. As an example, 5E3 stands for 3+(14X16)+(5X16X16) or 1507. E in base 16 stands for 14.

Computers store and read only binary code, but binary code is extremely difficult for us programmers to read and use. The counting of the 0's and 1's is very time consuming, and errors are bound to occur when the numbers you use get as long as 0010110101010111100000000111011110111. But a microcomputer memory works like a series of capacitors which can exist in one of two states, charged or discharged. That's why the binary number system is used in a computer.

To program in assembly language, binary code is often written in hexadecimal form so the programmer doesn't have to count 0's and 1's. Hexadecimal is chosen because four binary digits can be easily represented by one hexadecimal digit. Compare these two numbers and see which is easier to remember: 38AD or 0011100010101101. They both mean the same thing. The first is hexadecimal, the second binary.

Base Conversions will convert any positive number with a value less than 65535 from any of the three bases to another of the three. 65535 is the memory size of the Commodore 64 and the largest number that can be stored in 16 binary digits. The three bases are decimal, binary, and hexadecimal. Run this program and you'll see a list of the six functions that it will perform. This is called the menu because you make a selection just as you would if you were ordering food in a restaurant. You choose one function and type in the corresponding number, or type a 7 to end the program, then enter the value you would like to convert. When you hit RETURN you will get the answer and be returned, after touching another key, to the menu. You can then enter another number or, if you typed a 7 to end the program, the program will terminate and return you to the BASIC operating system.

Because the different computers store negative numbers in different forms I have not provided for negative numbers in this program.

There are four subroutines used that may be patched into any other program. They convert binary to decimal, decimal to hexadecimal, hexadecimal to decimal and decimal to binary. To perform the other functions I chain two subroutines together. For example, to convert a binary number to hexadecimal I first convert it to decimal, then convert the decimal number to hexadecimal. This is not the fastest way, but it saves memory.

If you thought that the mastery of base converting was an insurmountable obstacle in your programming education, we hope this program makes a convert out of you! □

SEE PROGRAM LISTINGS ON PAGE 96

PROGRAMS TYPED: $\frac{35¢}{HR.}$

Ahoy! PROGRAM DISK

We'll bet you're looking forward to trying out the fantastic programs in this issue of *Ahoy!* But we'll bet you're *not* looking forward to typing them in. If you're an average typist, that should take you around 23 hours—not counting debugging time.

How would you like someone to type the programs for you? For, say—35¢ an hour? Don't you think it would be worth 35¢ an hour to free yourself up for more pleasant pursuits—like enjoying the rest of your *Ahoy!* magazine?

If you order the *Ahoy! Program Disk*, you'll be getting that service for just under 35¢ an hour. Because for $7.95 we'll mail you all the programs in a given issue, on a disk that's tested and ready to run with your C-64 or VIC 20. If you subscribe, you'll save even more: 12 monthly disks for $79.95, 24 for $149.95.

You already know how to type. Why not use the hours you spend with your Commodore to learn something new?

- ☐ September Ahoy! Program Disk: $7.95 in USA, $10.00 in Canada. Elsewhere (outside USA and Canada) $12.00 Postage and handling included.
- ☐ 12-month Ahoy! Program Disk subscription: $79.95 in USA, $99.95 in Canada. Elsewhere (outside USA and Canada) $124.95 Postage and handling included.
- ☐ 24-month Ahoy! Program Disk subscription: $149.95 in USA, $179.95 in Canada. Postage and handling included.

NAME____________________

ADDRESS____________________

CITY__________ STATE______ ZIP______

Back issues are also available on disk:
January—$12.95; February—$11.95; March—$10.95; April—$9.95; May—$9.95; June—$8.95; July—$8.95; August—$8.95.
(In Canada add $2.00 per disk; outside USA and Canada add $4.00 per disk)

PROGRAM LISTINGS

On the following pages are listed several programs that we hope you'll want to punch in your Commodore computer. But please read the following introduction first; there are a few things you'll need to know.

Certain computer commands are displayed on the monitor by a variety of odd-looking characters. To get your computer to display these commands rather than actually perform them, you'll need to enter the quote mode. Hold down the SHIFT key and press the "2" key; a set of quote marks will appear. This tells the computer that the characters that follow are to be displayed, not performed. To exit the quote mode, type another set of quote marks, or hit the RETURN key. You'll also enter the quote mode when you INserT spaces or characters onto a line.

In *Ahoy!'s* program listings, you'll frequently find letters and/or numbers surrounded by brackets { }. That's because, for the purposes of clear reproduction, we at *Ahoy!* use a daisy wheel printer incapable of reproducing command symbols. For example, when you're in the quote mode and press the SHIFT and CLR/HOME keys at the same time, the screen (or a dot-matrix printer) will indicate this command with a heart { }. Because a daisy wheel cannot duplicate this symbol, it substitutes an alternate code between brackets. In the case of the SHIFT/CLR HOME symbol, our printer substitutes {SC}.

Another special case is SHIFT and COMMODORE characters. We represent these by underlining or overlining, respectively: any character underlined in the program listing should be punched in as a SHIFTed character ($\underline{J}$ = SHIFT J), any character overlined should be punched in as a COMMODORE character ($\overline{J}$ = COMMODORE J).

An alternate way of entering commands and other graphics symbols and characters is to use their corresponding character strings. The CLR/HOME command, for example, is entered by typing CHR$(147). While this requires a few extra strokes, it facilitates editing your program or reading the printed listing. For a complete list of CHR$ codes, consult the appendix at the back of your Commodore user manual.

Below is a list of the command abbreviations you'll find in our program listings, the commands they stand for, how to enter them, and how they'll appear on the screen or on a dot matrix printout.

When You See	It Means	You Type	You Will See
{SC}	Screen Clear	SHIFT CLR/HOME	
{HM}	Home	CLR/HOME	
{CU}	Cursor Up	SHIFT ↑ CRSR ↓	
{CD}	Cursor Down	↑ CRSR ↓	
{CL}	Cursor Left	SHIFT ← CRSR →	
{CR}	Cursor Right	← CRSR →	
{SS}	Shifted Space	SHIFT space	
{IN}	Insert	INST	
{RV}	Reverse On	CNTRL 9	
{RO}	Reverse Off	CNTRL 0	
{BK}	Black	CNTRL 1	
{WH}	White	CNTRL 2	
{RD}	Red	CNTRL 3	
{CY}	Cyan	CNTRL 4	
{PU}	Purple	CNTRL 5	
{GN}	Green	CNTRL 6	
{BL}	Blue	CNTRL 7	

When You See	It Means	You Type	You Will See
{YL}	Yellow	CNTRL 8	
{OR}	Orange	COMMODORE 1	
{BR}	Brown	COMMODORE 2	
{LR}	Light Red	COMMODORE 3	
{G1}	Grey 1	COMMODORE 4	
{G2}	Grey 2	COMMODORE 5	
{LG}	Light Green	COMMODORE 6	
{LB}	Light Blue	COMMODORE 7	
{G3}	Grey 3	COMMODORE 8	
{F1}	Function 1	F 1	
{F2}	Function 2	F 2	
{F3}	Function 3	F 3	
{F4}	Function 4	F 4	
{F5}	Function 5	F 5	
{F6}	Function 6	F 6	
{F7}	Function 7	F 7	
{F8}	Function 8	F 8	

IMPORTANT!
Before typing in the *Bug Repellent* and other *Ahoy!* programs, refer to the information on page 67.

VIC 20 BUG REPELLENT

By Michael Kleinert and David Barron

The program listed below will allow you to quickly debug any *Ahoy!* program you type in on your VIC 20. Follow directions for cassette or disk.

For cassette: type in and save the *Bug Repellent* program, then type RUN 63000[RETURN]SYS 828[RETURN]. If you typed the program properly, it will generate a set of two-letter line codes that will match those listed below the program on this page. (If you didn't type the program properly, of course, no line codes will be generated. You'll have to debug the *Bug Repellent* itself the hard way.)

Once you've got a working *Bug Repellent*, type in the program you wish to check. Save it and type the RUN and SYS commands listed above once again, then compare the line codes generated to those listed in the magazine. If you spot a discrepancy, a typing error exists in that line. Important: you must use exactly the same spacing as the program in the magazine. Due to memory limitations on the VIC, the VIC *Bug Repellent* will register an error if your spacing varies from what's printed.

You may type SYS 828 as many times as you wish, but if you use the cassette for anything, type RUN 63000 to restore the *Repellent.*

When your program has been disinfected you may delete all lines from 63000 on. (Be sure the program you type doesn't include lines above 63000!)

For disk: type in the *Bug Repellent*, save it, and type RUN:NEW[RETURN]. (See above regarding testing the *Bug Repellent* on itself.) Type in the program you wish to check, then SYS 828. This will generate a set of two-letter line codes that you should compare to those listed in the magazine.

To pause the line codes listing, press SHIFT. To permanently pause it, press SHIFT LOCK. To continue, release SHIFT LOCK.

To send the list to the printer type OPEN 4,4:CMD 4:SYS 828[RETURN]. When the cursor comes back, type PRINT#4:CLOSE 4[RETURN].

```
•63000 FOR X = 828 TO 1023 :READ Y
 :POKE X,Y:NEXT:END
•63001 DATA 169, 0, 133, 63, 133,
 64, 165, 43, 133, 251
•63002 DATA 165, 44, 133, 252, 160
 , 0, 132, 254, 32, 228
•63003 DATA 3, 234, 177, 251, 208,
  3, 76, 208, 3, 230
•63004 DATA 251, 208, 2, 230, 252,
  169, 244, 160, 3, 32
•63005 DATA 30, 203, 160, 0, 177,
 251, 170, 230, 251, 208
•63006 DATA 2, 230, 252, 177, 251,
  32, 205, 221, 169, 58
•63007 DATA 32, 210, 255, 169, 0,
 133, 253, 230, 254, 32
•63008 DATA 228, 3, 234, 165, 253,
  160, 0, 170, 177, 251
•63009 DATA 201, 32, 240, 6, 138,
 113, 251, 69, 254, 170
•63010 DATA 138, 133, 253, 177, 25
 1, 208, 226, 165, 253, 41
•63011 DATA 240, 74, 74, 74, 74, 2
 4, 105, 65, 32, 210
•63012 DATA 255, 165, 253, 41, 15,
  24, 105, 65, 32, 210
•63013 DATA 255, 169, 13, 32, 210,
  255, 173, 141, 2, 41
•63014 DATA 1, 208, 249, 230, 63,
 208, 2, 230, 64, 230
•63015 DATA 251, 208, 2, 230, 252,
  76, 74, 3, 169, 236
•63016 DATA 160, 3, 32, 30, 203, 1
 66, 63, 165, 64, 32
•63017 DATA 205, 221, 169, 13, 32,
  210, 255, 96, 230, 251
•63018 DATA 208, 2, 230, 252, 96,
 0, 76, 73, 78, 69
•63019 DATA 83, 58, 32, 0, 76, 73,
  78, 69, 32, 35
•63020 DATA 32, 0, 0, 0, 0, 0
```

BUG REPELLENT LINE CODES FOR VIC 20 BUG REPELLENT

```
LINE # 63000:MH     LINE # 63011:NN
LINE # 63001:BD     LINE # 63012:IG
LINE # 63002:FO     LINE # 63013:EN
LINE # 63003:ND     LINE # 63014:GJ
LINE # 63004:DJ     LINE # 63015:IK
LINE # 63005:LP     LINE # 63016:HG
LINE # 63006:JB     LINE # 63017:CK
LINE # 63007:JF     LINE # 63018:JF
LINE # 63008:KA     LINE # 63019:OH
LINE # 63009:HP     LINE # 63020:LH
LINE # 63010:KJ     LINES: 21
```

C-64 BUG REPELLENT

By Michael Kleinert and David Barron

The program listed below will allow you to quickly debug any *Ahoy!* program you type in on your C-64.

Type in, SAVE, and RUN the *Bug Repellent*, Type NEW, then type in or LOAD the *Ahoy!* program you wish to check. When that's done, SAVE your program (don't RUN it!) and type SYS 49152 [RETURN]. You'll be asked if you want the line value codes displayed on the screen or dumped to the printer. If you select screen, it will appear there.

The table will move quickly, too quickly for most mortals to follow. To pause the listing depress and hold the SHIFT key. To pause for an extended period, depress SHIFT LOCK. As long as it is locked, the display will remain frozen.

Compare the table your machine generates to the table in *Ahoy!* that follows the program you're entering: If you spot a difference, an error exists in that line. Jot down the numbers of lines where contradictions occur, LIST each line, spot the errors, and correct them.

```
•5000 FOR X = 49152 TO 49488 :READ
 Y:POKE X,Y:NEXT:END
•5001 DATA 32, 161, 192, 165, 43,
 133, 251, 165, 44, 133
•5002 DATA 252, 160, 0, 132, 254,
 32, 37, 193, 234, 177
•5003 DATA 251, 208, 3, 76, 138, 1
 92, 230, 251, 208, 2
•5004 DATA 230, 252, 76, 43, 192,
 76, 73, 78, 69, 32
•5005 DATA 35, 32, 0, 169, 35, 160
 , 192, 32, 30, 171
•5006 DATA 160, 0, 177, 251, 170,
 230, 251, 208, 2, 230
•5007 DATA 252, 177, 251, 32, 205,
 189, 169, 58, 32, 210
•5008 DATA 255, 169, 0, 133, 253,
 230, 254, 32, 37, 193
•5009 DATA 234, 165, 253, 160, 0,
 76, 13, 193, 133, 253
•5010 DATA 177, 251, 208, 237, 165
 , 253, 41, 240, 74, 74
•5011 DATA 74, 74, 24, 105, 65, 32
 , 210, 255, 165, 253
•5012 DATA 41, 15, 24, 105, 65, 32
 , 210, 255, 169, 13
•5013 DATA 32, 220, 192, 230, 63,
 208, 2, 230, 64, 230
•5014 DATA 251, 208, 2, 230, 252,
 76, 11, 192, 169, 153
•5015 DATA 160, 192, 32, 30, 171,
 166, 63, 165, 64, 76
•5016 DATA 231, 192, 96, 76, 73, 7
 8, 69, 83, 58, 32
•5017 DATA 0, 169, 247, 160, 192,
 32, 30, 171, 169, 3
•5018 DATA 133, 254, 32, 228, 255,
 201, 83, 240, 6, 201
•5019 DATA 80, 208, 245, 230, 254,
 32, 210, 255, 169, 4
•5020 DATA 166, 254, 160, 255, 32,
 186, 255, 169, 0, 133
•5021 DATA 63, 133, 64, 133, 2, 32
 , 189, 255, 32, 192
•5022 DATA 255, 166, 254, 32, 201,
 255, 76, 73, 193, 96
•5023 DATA 32, 210, 255, 173, 141,
 2, 41, 1, 208, 249
•5024 DATA 96, 32, 205, 189, 169,
 13, 32, 210, 255, 32
•5025 DATA 204, 255, 169, 4, 76, 1
 95, 255, 147, 83, 67
•5026 DATA 82, 69, 69, 78, 32, 79,
 82, 32, 80, 82
•5027 DATA 73, 78, 84, 69, 82, 32,
 63, 32, 0, 76
•5028 DATA 44, 193, 234, 177, 251,
 201, 32, 240, 6, 138
•5029 DATA 113, 251, 69, 254, 170,
 138, 76, 88, 192, 0
•5030 DATA 0, 0, 0, 230, 251, 208,
 2, 230, 252, 96
•5031 DATA 170, 177, 251, 201, 34,
 208, 6, 165, 2, 73
•5032 DATA 255, 133, 2, 165, 2, 20
 8, 218, 177, 251, 201
•5033 DATA 32, 208, 212, 198, 254,
 76, 29, 193, 0, 169
•5034 DATA 13, 76, 210, 255, 0, 0,
 0
```

BUG REPELLENT LINE CODES FOR C-64 BUG REPELLENT

LINE # 5000:GJ	LINE # 5018:FK
LINE # 5001:DL	LINE # 5019:FL
LINE # 5002:DB	LINE # 5020:CL
LINE # 5003:OF	LINE # 5021:GC
LINE # 5004:KN	LINE # 5022:NN
LINE # 5005:CA	LINE # 5023:NH
LINE # 5006:CE	LINE # 5024:IM
LINE # 5007:JE	LINE # 5025:KC
LINE # 5008:CL	LINE # 5026:DC
LINE # 5009:NB	LINE # 5027:ML
LINE # 5010:MB	LINE # 5028:GN
LINE # 5011:EP	LINE # 5029:JK
LINE # 5012:GH	LINE # 5030:NA
LINE # 5013:AN	LINE # 5031:DM
LINE # 5014:NG	LINE # 5032:JA
LINE # 5015:BF	LINE # 5033:FM
LINE # 5016:EP	LINE # 5034:PA
LINE # 5017:PJ	LINES: 35

DOS

FROM PAGE 21

```
10 PRINT "{SC}":POKE 53280,0:POKE
 53281,1:DIM AE$(200):PRINT
20 PRINT TAB(10);"{BL}{RV}£  "
25 PRINT TAB(10);"{RV} {RO}  {RV}
 {RO}£"
30 PRINT TAB(10);"{RV} {RO}  {RV}
 *{RO}OMMODORE C 64"
35 PRINT TAB(10);"*{RV}  "
40 FOR A=1 TO 3:PRINT:NEXT A
45 PRINT TAB(7);"{RD}D I S K"
50 PRINT TAB(11);"{CD}{RD}O P E R
 A T I N G"
55 PRINT TAB(21);"{CD}{RD} S Y S
 T E M"
60 FOR A=1 TO 4:PRINT:NEXT A
70 PRINT TAB(8);"{BL}DESIGNED BY.
..BOB LLORET"
80 PRINT TAB(11)"{CD}{PU}COPYRIGH
T {BK}(C) {PU}1984"
90 FOR D=1 TO 2500:NEXT D
100 PRINT "{SC}":POKE 53280,0:POK
E 53281,6
110 PRINT "{WH}{CD} UCCCC{RV} DIS
K OPERATING SYSTEM V3.0 {RO}CCCCI
 "
115 PRINT "{CU} B";SPC(36)"B"
116 PRINT " B";SPC(36)"B"
120 PRINT " B {RV}[1]{RO} FORMAT
DISK      {RV}[4]{RO} ERASE PRG B
"
125 PRINT " B";TAB(38)"B"
130 PRINT " B {RV}[2]{RO} RENAME
FILE      {RV}[5]{RO} WRITE DOS B
"
135 PRINT " B";TAB(38)"B"
140 PRINT " B {RV}[3]{RO} VALIDAT
E DISK    {RV}[6]{RO} DIRECTORY B
"
141 PRINT " B";TAB(38)"B"
142 PRINT " B              {RV}[7]{R
O} HELP SCREEN             B"
145 FOR A=1 TO 1:PRINT " B";TAB(3
8)"B":NEXT A
150 PRINT " JCCCCCCCC>YOUR CHOICE
 ?{RV}[1-7]{RO}<CCCCCCCCK":PRINT:
PRINT
160 GET AN$:IF AN$="" THEN 160
170 AN=VAL(AN$)
180 IF AN<1 OR AN>7 THEN 160
190 ON AN GOTO 200,300,400,500,60
0,700,1200
198 REM ****  FORMAT DISK  ****
199 REM ========================
200 PRINT TAB(8);"{RV}ENTER DISK
NAME (16 CHR)":INPUT "{CR}{CR}{CR
}{CR}{CR}{CR}{CR}{CR}{CR}{CR}";NA
$
210 PRINT TAB(10);"{RV}ENTER DISK
 ID (2 CHR)":INPUT "{CR}{CR}{CR}{
CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{C
R}{CR}{CR}{CR}";ID$
220 NA$="NEW0:"+NA$+","+ID$
230 PRINT TAB(10);"{BK}{RV} ARE Y
OU SURE (YNN) ?"
240 GET AN$:IF AN$="" THEN 240
250 IF AN$="Y" THEN 280
260 IF AN$="N" THEN 100
270 IF AN$<>"Y" OR AN$<>"N" THEN
240
280 OPEN 15,8,15,NA$
290 CLOSE 15
295 GOTO 100
298 REM ****  RENAME UTILITY  ***
*
299 REM =========================
=
300 PRINT TAB(8);"{RV}{WH} ENTER
NEW NAME (16 CHR)":INPUT "{CR}{CR
}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR}
{CR}{CR}";NN$
310 PRINT TAB(8);"{RV}{CD} ENTER
OLD NAME (16 CHR)":INPUT "{CR}{CR
}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR}
{CR}{CR}";OLN$
320 CH$="R0:"+NN$+"="+OLN$
330 OPEN 15,8,15,CH$:FOR D=1 TO 1
500:NEXT D
340 PRINT TAB(17);"{RV} DONE ":FO
R D=1 TO 1000:NEXT D
350 CLOSE 15:GOTO 100
398 REM ****  VALIDATE DISK  ****
399 REM =========================
400 PRINT TAB(12);"{WH}{RV}{CD} V
ALIDATING DISK "
410 OPEN 15,8,15,"VALIDATE"
440 CLOSE 15:GOTO 100
498 REM **** SCRATCH PROGRAM ****
499 REM =========================
500 PRINTTAB(3);"{WH}{RV}{CD} NAM
E OF PROGRAM TO ERASE (16 CHR)":I
NPUT "{CR}{CR}{CR}{CR}{CR}{CR}{CR
}{CR}{CR}{CR}{CR}";PN$
510 ER$="S0:"+PN$
```

```
520 OPEN 15,8,15,ER$
530 FOR D=1 TO 500:NEXT D:PRINT T
 AB(17);"{RV}{CD} DONE ":FOR D=1 T
 O 1500:NEXT D
540 CLOSE 15:GOTO 100
598 REM **** WRITE DOS ****
599 REM ===================
600 PRINT TAB(9);"{WH}{RV}{CD} WR
 ITING D O S FILES "
610 SAVE "DOS",8
620 PRINT TAB(17);"{RV}{CD} DONE
 ":FOR D=1 TO 1500:NEXT D
630 CLOSE 15:GOTO 100
700 REM ****  DISK DIRECTORY  ***
 *
705 REM =========================
 =
710 PRINT "{SC}":POKE 53280;4:POK
 E 53281,1
720 AE$="":AN=0:A0=0:DE=0:DR$="0"
730 ER=0:F$="":FL=0:I=0:J=0:MM=0:
 MN=0
750 OPEN 15,8,15:PRINT#15,"I"+DR$
760 INPUT#15,ER:IF ER=21 THEN890
770 OPEN 8,8,8,"$"+DR$+",SEQ"
780 FOR DE=1 TO 8:F$="":GET #8,C$
790 IF C$=CHR$(199) THENCLOSE 8:C
 LOSE 15:GOTO900
800 IF C$="" THENJ=29:GOTO860
810 IF ASC(C$)<>130 THENJ=29:GOTO
 860
820 AN=AN+1:J=11:GET #8,C$:GET #8
 ,C$
830 FOR I=1 TO 16:GET #8,C$:F$=F$
 +C$:NEXT
840 IF LEFT$(F$,3)="DOS" THENAN=A
 N-1:GOTO860
850 AE$(AN)=DR$+":"+F$
860 FOR I=1 TO J:GET #8,C$:NEXT
870 IF DE<>8 THENGET #8,C$:GET #8
 ,C$
880 NEXT:GOTO780
890 PRINT"{CD}{CD}{CD}{CD}{CD}{RV
 }NO DISKETTE FOUND IN DRIVE";DR$;
 "{CD}":FORD=1TO 2500:NEXT:GOTO 10
 0
900 IF AN=0 THENPRINT "{BK}{CD}{C
 D}{CD}{CD}{CD}{CR}{CR}{RV}NO PROG
 RAMS FOUND {RO}":FORD=1TO2500:NEX
 T:GOTO 100
910 PRINT "{SC}"TAB(7)"{CD}{PU}UC
 CCCCCCCCCCCCCCCCCCCCCCI"
920 MM=9:PRINT TAB(7)"B * D I R E
 C T O R Y * B"
930 PRINT TAB(7)"{PU}JCCCCCCCCCCC
 CCCCCCCCCCCCK":PRINT "{CD}{CD}{CD
 }"
940 FOR I=1 TO 9:IF AE$(MN*9+1)="
 " THENMM=I-1:I=9:GOTO960
950 PRINT TAB(11);"{PU}{RV}[";RIG
 HT$(STR$(I),1);"]{BL}{RO} ";MID$(
 AE$(MN*9+I),3,16)
960 NEXT:PRINT TAB(11)"{PU}{CD}{C
 D}{CD}YOUR CHOICE [1-9]"
965 PRINT TAB(11)"PRESS F1 FOR DO
 S"
970 PRINT TAB(11)"OR RETURN FOR M
 ORE"
980 GET C$:IF C$="" THEN980
985 IF C$="{F1}" THEN 100
990 IF C$<>CHR$(13) THEN1020
1000 MN=MN+1:IF MN*9+1>AN THENMN=
 0
1010 GOTO910
1020 IF VAL(C$)<1 OR VAL(C$)>MM T
 HEN980
1030 AE$=AE$(MN*9+VAL(C$))
1040 PRINT:PRINT "{SC}{CD}{CD}{CD
 }{CD}MENU ITEM CHOSEN: #";C$;" -
 ";MID$(AE$,3,16)
1050 FOR I=18 TO 1 STEP -1:FL=I
1060 IF ASC(MID$(AE$,I,1))<>160 T
 HENI=1
1070 NEXT:PRINT "{CD}{CD}{CD}{CD}
 LOAD";CHR$(34);LEFT$(AE$,FL);CHR$
 (34);",8{CD}{CD}{CD}{CD}"
1080 PRINT "RUN":PRINT "{CU}{CU}{
 CU}{CU}{CU}{CU}{CU}{CU}{CU}"
1090 POKE 631,13:POKE 632,13:POKE
  198,2:END
1200 PRINT"{SC}":POKE 53280,0:POK
 E 53281,0
1210 PRINT TAB(7)"{PU} * H E L P
  S C R E E N *"
1220 PRINT"{RD}{CD}{RV}[1]{RO}  P
 RESS #1{LB} TO FORMAT A NEW DISK
 OR       ANY DISK YOU WISH ";
1230 PRINT "TO TOTALY ERASE."
1240 PRINT "{RV}{RD}[2]{RO}  PRES
 S #2{LB} TO CHANGE THE NAME OF AN
 Y    PROGRAM ON A DISK.";
1250 PRINT " YOU SHOULD ALWAYS
  LIST THE DISK DIRECTORY FIRST TO
  MAKE   SURE";
1260 PRINT " YOU SPELL THE OLD NA
 ME CORRECTLY   TO INSURE NO DISK
```

```
ERRORS."
•1270 PRINT "{RV}{RD}[3]{RO}  PRESS #3{LB} AFTER YOU HAVE USED A DISK   FOR A WHILE. THIS ";
•1280 PRINT "WILL COMPACT YOUR  PROGRAMS AND ELIMINATE UNUSED BLOCKS ";
•1290 PRINT "   ON A DISK."
•1300 PRINT "{RV}{RD}[4]{RO}  PRESS #4{LB} WHEN YOU WISH TO ERASE ANY   PROGRAM ON A DISK.";
•1310 PRINT" ALWAYS LIST THE DISK DIRECTORY TO INSURE PROPER SPELL   ING";
•1320 PRINT " OF THE PROGRAM."
•1330 PRINT "{RV}{RD}[5]{RO}  PRESS #5{LB} TO MAKE A COPY OF D O S ON   ALL YOUR DISKS.";
•1340 PRINT " THIS WILL MAKE IT     HANDY IF YOU EVER NEED IT."
•1350 PRINT "{RD}{RV}[6]{RO}  PRESS #6{LB} TO LIST THE DIRECTORY AND    RUN ANY PROGRAM";
•1355 PRINT " ON THE DISK."
•1360 PRINT TAB(10)"{YL}{CD}<PRESS F1 FOR D O S>";
•1370 GET AN$:IF AN$="" THEN 1370
•1380 IF AN$="{F1}" THEN 100
•1390 IF AN$<>"{F1}" THEN 1370
```

BUG REPELLENT LINE CODES FOR DOS

```
LINE # 10:CK
LINE # 20:EJ
LINE # 25:NI
LINE # 30:EK
LINE # 35:MB
LINE # 40:GN
LINE # 45:MJ
LINE # 50:BB
LINE # 55:BB
LINE # 60:FM
LINE # 70:DG
LINE # 80:II
LINE # 90:BI
LINE # 100:OJ
LINE # 110:EK
LINE # 115:MC
LINE # 116:FN
LINE # 120:FM
LINE # 125:DI
LINE # 130:KE
LINE # 135:DI
LINE # 140:NO
LINE # 141:DI
LINE # 142:EP
LINE # 145:BO
LINE # 150:CD
LINE # 160:CJ
LINE # 170:BK
LINE # 180:HH
LINE # 190:NE
LINE # 198:CC
LINE # 199:CD
LINE # 200:FP
LINE # 210:DD
LINE # 220:ME
LINE # 230:HN
LINE # 240:EM
LINE # 250:OD
LINE # 260:LH
LINE # 270:OJ
LINE # 280:OP
LINE # 290:AB
LINE # 295:CF
LINE # 298:EJ
LINE # 299:ME
LINE # 300:MM
LINE # 310:FI
LINE # 320:LD
LINE # 330:II
LINE # 340:NJ
LINE # 350:FN
LINE # 398:CI
LINE # 399:JO
LINE # 400:II
LINE # 410:MK
LINE # 440:FN
LINE # 498:NK
LINE # 499:JO
LINE # 500:FC
LINE # 510:PO
LINE # 520:PL
LINE # 530:IM
LINE # 540:FN
LINE # 598:IG
LINE # 599:CH
LINE # 600:JI
LINE # 610:CD
LINE # 620:CJ
LINE # 630:FN
LINE # 700:OD
LINE # 705:ME
LINE # 710:HO
LINE # 720:HH
LINE # 730:KN
LINE # 750:JL
LINE # 760:IC
LINE # 770:BM
LINE # 780:AD
LINE # 790:HA
LINE # 800:DE
LINE # 810:JG
LINE # 820:EB
LINE # 830:GM
LINE # 840:BA
LINE # 850:CK
LINE # 860:AN
LINE # 870:MO
LINE # 880:OK
LINE # 890:IL
LINE # 900:LJ
LINE # 910:JD
LINE # 920:MJ
LINE # 930:NK
LINE # 940:GN
LINE # 950:PG
LINE # 960:HG
LINE # 965:AE
LINE # 970:KL
LINE # 980:HI
LINE # 985:KE
LINE # 990:CD
LINE # 1000:DC
LINE # 1010:CK
LINE # 1020:IC
LINE # 1030:PL
LINE # 1040:CD
LINE # 1050:CF
LINE # 1060:GG
LINE # 1070:DN
LINE # 1080:DP
LINE # 1090:AN
LINE # 1200:OD
LINE # 1210:HL
LINE # 1220:MH
LINE # 1230:BG
LINE # 1240:GP
LINE # 1250:PK
LINE # 1260:NI
LINE # 1270:CL
LINE # 1280:OP
LINE # 1290:KK
LINE # 1300:IF
LINE # 1310:NM
LINE # 1320:CF
LINE # 1330:CC
LINE # 1340:HD
LINE # 1350:FB
LINE # 1355:BI
LINE # 1360:OL
LINE # 1370:FA
LINE # 1380:AO
LINE # 1390:PF
LINES: 132
```

SALVAGE DIVER

FROM PAGE 23

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
1 STOP
2 POKE53280,7:POKE53281,0
4 PRINT"{SC}{WH}{CD}{CD}{CD}{CD}"
;SPC(7);"{RV}***** SALVAGE DIVER!
 *****"
6 PRINT"{CD}{CD}{CD}";SPC(13);"A
C-64 PROGRAM"
7 PRINT"{CD}{CD}";SPC(11);"(C) CO
PYRIGHT 1984"
8 PRINT"{CD}{CD}";SPC(13);"BY B.W
.BEHLING"
10 PRINTSPC(3);"{CD}{CD}{CD}{CD}
PLEASE WAIT. READING DATA......."
12 FORB=0TO4:FORI=0TO62
14 READA:POKE(240+B)*64+I,A
16 NEXTI:NEXTB
18 DIMT$(56),T(56,1):FORI=0TO56:R
EADW$:READW:T$(I)=W$:T(I,0)=W:T(I
,1)=0:NEXTI
20 FORI=49152TO49301:READA:POKEI,
A:NEXTI
22 FORI=49408TO49575:READA:POKEI,
A:NEXTI:RESTORE
24 FORI=0TO62:POKE246*64+I,0:NEXT
I
30 MA=15766:POKEMA,195:POKEMA+3,1
02:POKEMA+6,60:POKEMA+9,24:POKEMA
+12,60
32 POKEMA+15,102:POKEMA+18,195
34 FORI=0TO62STEP3:FORN=2TO0STEP-
1
36 READA:GOSUB100
38 POKE245*64+(I+N),A2:NEXTN:NEXT
I
40 REM***** START *****
50 WD=0:WL=0:WD$="0.00":WL$="0.00
":BA=0:DI=3:DD=0:DA=2000:DA$="200
0"
52 V=53248:M=2040:POKEV+21,0:POKE
V+38,7:POKEV+37,10:POKEV+28,128:S
1=1024
54 S2=1944:POKEV+27,0:POKEM+6,246
:K=54272:C1=55296:C2=56216
56 FORI=39TO44:POKEV+I,11:NEXTI:P
OKEV+45,1:POKEV+46,4
58 FORI=0TO10STEP2:POKEV+I,0:NEXT
I:POKEV+16,0:GOTO600
60 REM***** DRAW SCREEN *****
61 PRINT"{SC}":POKE53280,6:FORI=0
TO79:POKES1+I,160:POKEC1+I,1:NEXT
I
62 FORI=80TO159:POKES1+I,160:POKE
C1+I,14:NEXTI
64 FORI=0TO80:POKES2+I,160:POKEC2
+I,12:NEXTI
66 PRINT"{HM}{CD}{CD}{CD}{CD}{CD}
{CD}{CD}{CD}{CD}{GN} Y";SPC(15);"
{BR}B"
68 PRINT"{GN} Y";SPC(12);"Y {BR}C
+C        {GN}Y";SPC(12);"Y"
70 PRINT"{GN} Y";SPC(12);"Y  {BR}
Q{G2}CCCCCCCCCC{BR}W";SPC(8);"{G
N}Y""
72 PRINT"{GN} Y";SPC(12);"Y  {BR}
B";SPC(7);"{GN}Y   {BR}B {GN}Y
  Y"
74 PRINT"{GN} Y";SPC(12);"Y{BR}CC
+CC       {GN}Y  {BR}C+C{GN}Y
Y"
76 PRINT"{GN} Y  Y";SPC(9);"Y  {B
R}B     {BL}{RV}     {RO}{GN}Y   {BR
}B {GN}Y     Y Y"
78 PRINT"{GN} Y  Y";SPC(9);"Y  {B
R}B     {RD}{RV}     {RO}{GN}Y   {BR
}B {GN}Y Y   Y Y"
80 PRINT"{GN} Y  Y     Y   Y {BR
}B  {G3}↑ {RV}     {RO}{GN}Y{G3}↑
{BR}B {GN}Y Y   Y Y"
82 PRINT"{GN} Y  Y     Y  {G3}↑{G
N}Y  {BR}B{GN}Y {G3}{RV}  TTT  {R
O}  {BR}B {GN}Y Y  Y Y"
84 PRINT"{GN} Y  Y     Y {G3}{RV}
Q Q{RO}   {BR}B{GN}Y {G3}{RV} Q Q
Q {RO}   {BR}B RRRR {GN}Y Y"
86 PRINT"{GN} Y YY{G3}*{RV}  *{RO}
{BR}RRRR{RV}{G3}      {RO}{BR}RR±RR{
RV}{G3}          {RO}{BR}RR±R{RV}{G3
}     {RO}{GN} Y Y"
88 PRINT"{GN} Y YY {G3}*{RV}S.S.
TTT        TTTTTTT Q    Q  {RO}{GN} Y
 Y"
90 PRINT"{GN} Y YY   {G3}*{RV}MARI
E  Q Q     Q Q         {RO}£B {GN}Y
 Y"
92 PRINT"{GN} Y YY    {BR}*{RV}
                      {RO}£{GN}Y{BR
}{RV} {RO}{GN} Y Y{HM}"
99 RETURN
100 REM***** SHARK REVERSE DATA *
****
102 IFA=255ORA=0THENA2=A:RETURN
104 A2=0:FORP=7TO0STEP-1:Z=INT(A/
2↑P)
106 IFZ=0THENGOTO110
108 A2=A2+2↑(7-P):A=A-(Z*2↑P)
110 NEXTP:RETURN
```

```
150 REM** UPDATE SHARK Y **
152 YU=INT(RND(0)*240):IFYU<170TH
EN152
154 POKEV+((2*D)+1),YU
156 RETURN
200 REM***** UPDATE SHARK X *****
202 D=INT(RND(1)*6):POKE49395,D:P
OKE49396,D*2
204 POKE49393,(INT(RND(0)*5)):POK
E49394,(5+QQ)
206 POKE49397,(2↑D):POKE49398,(25
5-2↑D)
208 SYS49152:SYS49408:GOSUB350:SY
S49152
210 SG=PEEK(V+30):SG=PEEK(V+30):I
FSG>128ANDSG<192THENGOSUB430:GOTO
218
212 IFSG>64ANDSG<128THENPOKEV+21,
191:GOSUB402:GOSUB550
214 CY=PEEK(V+16):CY%=2↑D:IFCY=CY
%ANDPEEK(V+(2*D))>88THENGOSUB150
216 RETURN
218 PRINT"{HM}{CD}{CD}{CD}";TAB(1
2);"{WH}{RV} SHARK GOT YOU! {RO}"
:DD=DD+1:POKEK+24,0
220 FORI=0TO3000:NEXTI
222 PRINT"{HM}{CD}{CD}{CD}";TAB(1
2);"{LB}{RV}                 {RO}"
:GOTO450
300 REM***** DIVER FINISHED *****
302 IFBA<=0THENRETURN
304 WL=WL+WD:WL$=STR$(WL):WD=0:WD
$="0.00":BA=0:BA$="0"
306 PRINT"{HM}";TAB(33-LEN(WL$));
"{WH}{RV} NET $";WL$
308 PRINT"{HM}{CD}{WH}{RV} TREASU
RE # 0           DIVER HAS $0.00"
310 PRINT"{CD}";SPC(14);"{WH}{RV}
 GOOD WORK!! "
312 FORI=0TO3000:NEXTI
314 PRINT"{HM}{CD}{CD}{CD}";TAB(1
2);"{LB}{RV}                 {RO}"
:GOTO450
350 REM***** UPDATE TIMER *****
352 IFPEEK(V+15)<81THENGOSUB300
354 DA=DA-(INT(RND(0)*2)+1)
356 QS=QS+1:IFQS>150THENQS=0:QQ=Q
Q+2
358 IFDA<=0THENPRINT"{HM}{WH}{RV}
 OUT OF AIR!! ":DD=DD+1:GOTO378
360 DA$=STR$(DA)
362 IFDA<100THENBK=BK+1:IFBK<>2TH
ENBK=1
364 IFDA<100THENGOSUB420:SYS49152
:ONBKGOTO370,374
366 IFDA<200THEN374
368 IFDA<500THEN372
370 PRINT"{HM}{WH}{RV} AIR";DA$;"
 LBS. ":GOTO376
372 PRINT"{HM}{YL}{RV} AIR";DA$;"
 LBS. ":GOTO376
374 PRINT"{HM}{RD}{RV} AIR";DA$;"
 LBS.{WH} "
376 RETURN
378 FORI=0TO3000:NEXTI:POKEV+21,6
3:GOTO450
400 REM***** SOUNDS *****
402 REM** BEEP **
404 POKEK+1,20:POKEK+5,9:POKEK+6,
248:POKEK+24,15:POKEK+4,17
406 FORU=1TO40:NEXTU:POKEK+4,16:F
ORI=0TO24:POKEK+I,0:NEXTI:RETURN
408 REM** GONG **
410 FORI=0TO22:POKEK+I,0:NEXTI
412 POKEK+24,143:POKEK+5,16:POKEK
+19,16
414 POKEK+6,252:POKEK+20,249:POKE
K+4,21:POKEK+18,17
416 POKEK+1,68:POKEK+15,42:FORI=1
TO200:NEXTI
418 POKEK+4,20:POKEK+18,16:FORI=1
TO400:NEXTI:RETURN
420 REM** ALARM **
422 FORI=0TO22:POKEK+I,0:NEXTI
424 POKEK+24,6:POKEK+5,80:POKEK+6
,243:POKEK+3,4:POKEK+4,65:SYS4915
2
426 FORI=0TO100STEP4:POKEK+1,I:NE
XTI:POKEK+4,64:FORI=1TO50:NEXTI:P
OKEK+4,64
428 RETURN
430 REM** SHARK ATTACK **
432 FORI=0TO22:POKEK+I,0:NEXTI
434 POKEK+24,15:POKEK+5,80:POKEK+
6,243:POKEK+3,4:POKEK+4,129
436 FORI=0TO140STEP4:POKEK+1,I:NE
XTI:POKEK+4,128:FORI=1TO50:NEXTI:
POKEK+4,128
438 RETURN
450 REM***** START DIVER *****
452 BA=0:WD=0:WD$="0.00":IFDI<=0T
HEN650
454 DA=2000:DA$=STR$(DA):QQ=0:QS=
0
456 DY=80:POKEV+15,DY:POKE(V+16),
PEEK(V+16)AND127:POKEV+21,255
```

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
•458 FORXD=0TO170:DN=DN+1:IFDN>1TH
 ENDN=0
•460 POKEM+7,243+DN:POKEV+14,XD:NE
 XTXD:DI=DI-1
•462 PRINT"{HM}{WH}{RV} AIR";DA$;"
  LBS. ";TAB(15);"{RV}{GN}DIVERS";
 DI
•464 PRINT"{HM}";TAB(31-LEN(WL$));
 "{WH}{RV}   NET $";WL$
•466 BA$="0":PRINT"{HM}{CD}{WH}{RV
 } TREASURE # 0 "
•468 PRINT"{HM}{CD}"TAB(24-LEN(WD$
 ));"{WH}{RV}     DIVER HAS $";WD$:
 GOTO550
•500 REM***** DETERMINE TREASURE *
 ****
•502 GT=PEEK(V+30)
•504 IFGT>191THENPOKE(V+21),191:BA
 =BA+1:GOSUB408:GOTO508
•506 RETURN
•508 T%=INT(RND(0)*112):IFT%>56THE
 NT%=0:GOTO514
•510 IFT(T%,1)<>0THEN508
•512 IFT%>11THENT(T%,1)=1
•514 BA$=STR$(BA):PRINT"{HM}{CD}{W
 H}{RV} TREASURE #";BA$;" "
•516 PRINT"{HM}{CD}{CD}{WH}{RV} YO
 U FOUND ";T$(T%);"!"
•518 PRINT"{RV} VALUE $";T(T%,0):I
 FT(T%,0)=0THEN522
•520 WD=VAL(WD$)+T(T%,0):WD$=STR$(
 WD)
•522 FORL=0TO1500:NEXTL:FORI=0TO22
 :POKEK+I,0:NEXTI
•524 FORI=80TO159:POKES1+I,160:POK
 EC1+I,14:NEXTI
•526 PRINT"{HM}{CD}"TAB(24-LEN(WD$
 ));"{RV}     DIVER HAS $";WD$
•550 REM***** MARK TREASURE *****
•552 POKE(V+16),PEEK(V+16)AND191
•554 BY=INT(RND(0)*(190-236)+236):
 POKEV+13,BY
•556 XM=INT(RND(0)*(24-255)+255):P
 OKEV+12,XM:POKEM+6,246:POKEV+21,2
 55:RETURN
•600 REM***** MAIN PROGRAM *****
•602 FORI=0TO5:POKEM+I,240:NEXTI
•604 FORI=0TO10STEP2:RS=INT(RND(0)
 *255):POKEV+I,RS:NEXTI
•606 FORD=0TO6:GOSUB150:NEXTD:GOSU
 B60:POKEV+21,63
•608 PRINT"{HM}{WH}{RV} AIR ";DA$;
 " LBS. ";TAB(15);"{RV}{GN}DIVERS"
 ;DI
•610 PRINT"{HM}";TAB(31-LEN(WL$));
 "{WH}{RV}   NET $";WL$
•612 BA$="0":PRINT"{HM}{CD}{WH}{RV
 } TREASURE # 0";TAB(27-LEN(WD$));
 " DIVER HAS $";WD$
•614 GOSUB450
•616 GOSUB200:GOSUB500:SYS49152
•618 GOTO616
•650 REM***** GAME OVER *****
•652 POKEV+21,0:PRINTCHR$(147):POK
 E53280,0
•654 PRINTSPC(9);"{WH}THE SEA HAS
 WON AGAIN!"
•656 PRINT"{CD}{CD}";SPC(6);"HOWEV
 ER, YOU HAVE RECOVERED;"
•658 PRINT"{CD}{RV}";TAB(19-INT(LE
 N(WL$)/2));"$";WL$
•660 PRINT"{CD}";SPC(6);"WORTH OF
 ASSORTED TREASURE!!"
•662 IFDD=0THEN670
•664 IFDD=1THENPRINT"{CD}{CD}   UN
 FORTUNATELY, YOU LOST A DIVER.":F
 E=12495:GOTO668
•666 PRINT"{CD}{CD}";SPC(3);"UNFOR
 TUNATELY, YOU LOST";DD;"DIVERS.":
 FE=DD*12495
•668 PRINT"{CD}{CD}";SPC(2);"FUNER
 AL EXPENSES WILL COST {RV}$";FE;"
 {CL}."
•670 NT$=STR$(VAL(WL$)-FE):IFVAL(N
 T$)<=0THEN676
•672 PRINT"{CD}{CD}";TAB(13);"YOUR
  PROFIT IS"
•674 PRINT"{CD}{RV}";TAB(19-INT(LE
 N(NT$)/2));"$";NT$:GOTO682
•676 IFVAL(NT$)=0THENPRINT"{CD}{CD
 }";SPC(8);"YOU DIDN'T MAKE A DIME
 !!":GOTO682
•678 PRINT"{CD}{CD}";TAB(14);"YOUR
  LOSS IS"
•680 PRINT"{CD}{RV}";TAB(19-INT(LE
 N(NT$)/2));"$";NT$
•682 PRINT"{CD}{CD}{CD}";SPC(12);"
 PLAY AGAIN (Y/N)?"
•684 GETK$:IFK$=""THEN684
•686 IFK$="Y"THENRUN10
•688 END
•700 REM********** DATA **********
•702 REM***** SHARK DATA *****
•704 DATA0,0,0,0,0,0,0,16,0,0,48,0
 ,0,120,1,0,254,7,127,255,206,223,
 255,252
```

```
706 DATA255,255,248,7,255,156,252
,2,14,0,1,3,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0
708 DATA0,0,0,0,0,0,0,0,0,0,0,0
710 REM***** DIVER SWIMMING 1 ***
**
712 DATA0,0,128,0,0,0,0,8,0,0,128
,0,0,0,0,2,0,0,40,255,0,43,255,8,
26,138,33
714 DATA10,138,128,8,138,128,96,0
,41,0,0,0,0,0,0,0,0,0
716 DATA0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0
718 REM***** DIVER SWIMMING 2 ***
**
720 DATA0,32,0,0,136,0,0,0,0,0,12
8,0,0,0,0,2,0,0
722 DATA40,255,0,43,255,0,26,138,
41,10,138,128,8,138,128,32,0,32
724 DATA64,0,8,0,0,1,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
726 REM***** DIVER SWIMMING 3 ***
**
728 DATA2,0,0,0,0,0,0,32,0,0,2,0,
0,0,0,0,0,128,0,255,40,32,255,232
,72,162,164
730 DATA2,162,160,2,162,32,104,0,
9,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
,0
732 DATA0,0,0,0,0,0,0,0,0,0,0
734 REM***** DIVER SWIMMING 4 ***
**
736 DATA2,0,0,0,0,0,0,32,0,0,2,0,
0,0,0,0,0,128,0,255,40,0,255,232,
104,162,164
738 DATA2,162,160,2,162,32,8,0,8,
32,0,1,64,0,0,0,0,0
740 DATA0,0,0,0,0,0,0,0,0,0,0,0,0
,0,0,0,0,0
742 REM***** TREASURE DATA *****
744 DATA"A ROCK",0,"A GOLD CHAIN"
,800,"A SILVER BAR!",300,"A GOLD
BAR!",1200
746 DATA"SOME GOLD COINS!",1500,"
SOME SILVER COINS!",235
748 DATA"SOME LOBSTERS!",60,"A BE
ER CAN",0,"AN OLD TIRE",0
750 DATA"A HUMAN SKELETON!",0,"AN
 OLD SWORD",15,"A SILVER TRAY!",3
00
752 DATA"A DIAMOND RING!",1300,"A
 SOLID GOLD STATUE!!",9000
754 DATA"A RUBY CHALICE!",2200,"A
N EMERALD BRACELET!",3200
756 DATA"A TIMEX WATCH.....IT TIC
KS!",2,"A BOX OF JEWELRY!",8765
758 DATA"A PLATINUM GOBLET!",5600
,"A '67 CHEVY ENGINE!",35,"A KITC
HEN SINK",6
760 DATA"A WW II DUD TORPEDO!",0,
"A PAIR OF SHOES IN CEMENT!",0
762 DATA"A LARGE CHEST OF GEMS!!"
,135742,"A PHONE BOOTH!",0
764 DATA"A BROKEN COMPUTER",0,"AN
 ANTIQUE SILVER CROSS!",795,"A TO
OL BOX",18
766 DATA"A SILVER CANDELABRA!",98
75,"A CASE OF RARE BRANDY!",3200
768 DATA"A BARBERS CHAIR",65,"A J
ADE THRONE!!",237865,"A BRASS BED
",150
770 DATA"A JEWELED CROWN!!",12733
5,"A SIGN 'ATLANTIS 15 KM.'!",0
772 DATA"A GOLD SCEPTER!",2400,"A
 DIAMOND TIARA!",56895,"A GOLD SH
IELD!",4300
774 DATA"A JEWELED DAGGER",1695,"
AN OUTBOARD MOTOR",95,"AN OLD RAD
IATOR",0
776 DATA"A JUNKED VOLKSWAGON",0,"
A WW II FIGHTER PLANE!",3400,"A S
TEEL DESK",0
778 DATA"A DENTED COFFEE POT",0,"
A BRASS COMPASS",135,"A POOL TABL
E",35
780 DATA"SOME SILVER CANDLESTICKS
",450,"AN OLD SODA MACHINE",23
782 DATA"A CAST IRON STOVE",0,"A
BOX OF BRASS LOCKS",85,"AN EMPTY
COFFIN!",0
784 DATA"A SAPPHIRE PENDANT!",130
0,"A RUSTED SUIT OF ARMOR",75
786 DATA"A CRYSTAL VASE",40,"A CR
ATE OF DOORKNOBS",65,"A SHIPS WHE
EL",0
788 REM***** JOYSTICK ML DATA ***
**
790 DATA173,0,220,160,80,162,236,
74,176,8,204,15,208,240,3,206
792 DATA15,208,74,176,8,236,15,20
8,240,3,238,15,208,74,176,47
794 DATA173,14,208,208,31,173,16,
208,41,128,208,16,173,16,208,9
796 DATA128,141,16,208,169,80,141
,14,208,96,234,234,173,16,208,41
798 DATA127,141,16,208,206,14,208
```

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
,162,241,32,133,192,96,234,234,74
•800 DATA176,37,162,243,32,133,192
,238,14,208,240,30,169,80,205,14
•802 DATA208,208,20,173,16,208,41,
128,240,13,173,16,208,41,127,141
•804 DATA16,208,169,0,141,14,208,9
6,234,234,173,16,208,9,128,141
•806 DATA16,208,96,234,234,236,255
,7,208,4,238,255,7,96,142,255
•808 DATA7,96,234,234,0,0
•810 REM***** SHARK X ML DATA ****
*
•812 DATA169,0,162,0,160,0,206,242
,192,173,241,192,240,72,169,240
•814 DATA172,243,192,153,248,7,174
,244,192,189,0,208,208,42,173,16
•816 DATA208,45,245,192,208,25,173
,16,208,13,245,192,141,16,208,169
•818 DATA90,174,244,192,157,0,208,
173,242,192,208,202,96,0,0,173
•820 DATA16,208,45,246,192,141,16,
208,174,244,192,222,0,208,173,242
•822 DATA192,208,179,96,0,0,169,24
5,172,243,192,153,248,7,174,244
•824 DATA192,254,0,208,240,46,169,
90,174,244,192,221,0,208,208,25
•826 DATA173,16,208,45,245,192,240
,17,173,16,208,45,246,192,141,16
•828 DATA208,169,0,174,244,192,157
,0,208,173,242,192,240,3,76,6
•830 DATA193,96,0,0,173,16,208,13,
245,192,141,16,208,173,242,192
•832 DATA240,3,76,6,193,96,0,0
```

BUG REPELLENT LINE CODES FOR SALVAGE DIVER

```
LINE # 1:JC
LINE # 2:CE
LINE # 4:DC
LINE # 6:LO
LINE # 7:PM
LINE # 8:OD
LINE # 10:GM
LINE # 12:JP
LINE # 14:MM
LINE # 16:MP
LINE # 18:IL
LINE # 20:MI
LINE # 22:HI
LINE # 24:EL
LINE # 30:AH
LINE # 32:MD
LINE # 34:MJ
LINE # 36:CN
LINE # 38:CE
LINE # 40:LK
LINE # 50:FJ
LINE # 52:NF
LINE # 54:KM
LINE # 56:FA
LINE # 58:GP
LINE # 60:AN
LINE # 61:NI
LINE # 62:AP
LINE # 64:LL
LINE # 66:DC
LINE # 68:GP
LINE # 70:JF
LINE # 72:LF
LINE # 74:AM
LINE # 76:AC
LINE # 78:FP
LINE # 80:BN
LINE # 82:PB
LINE # 84:BN
LINE # 86:KJ
LINE # 88:KD
LINE # 90:AM
LINE # 92:GC
LINE # 99:IM
LINE # 100:PH
LINE # 102:OI
LINE # 104:OI
LINE # 106:KJ
LINE # 108:OF
LINE # 110:JP
LINE # 150:OO
LINE # 152:PJ
LINE # 154:DB
LINE # 156:IM
LINE # 200:CM
LINE # 202:EF
LINE # 204:HE
LINE # 206:GK
LINE # 208:LE
LINE # 210:GO
LINE # 212:OB
LINE # 214:HD
LINE # 216:IM
LINE # 218:IA
LINE # 220:DB
LINE # 222:BM
LINE # 300:LA
LINE # 302:PB
LINE # 304:JB
LINE # 306:JG
LINE # 308:NK
LINE # 310:AG
LINE # 312:DB
LINE # 314:BM
LINE # 350:KP
LINE # 352:NG
LINE # 354:JE
LINE # 356:DN
LINE # 358:OM
LINE # 360:PH
LINE # 362:IM
LINE # 364:JB
LINE # 366:KF
LINE # 368:KG
LINE # 370:FB
LINE # 372:LL
LINE # 374:LJ
LINE # 376:IM
LINE # 378:FD
LINE # 400:CN
LINE # 402:EN
LINE # 404:CM
LINE # 406:NN
LINE # 408:HK
LINE # 410:DG
LINE # 412:AN
LINE # 414:GH
LINE # 416:DC
LINE # 418:LG
LINE # 420:KC
LINE # 422:DG
LINE # 424:IL
LINE # 426:ID
LINE # 428:IM
LINE # 430:KJ
LINE # 432:DG
LINE # 434:HJ
LINE # 436:JK
LINE # 438:IM
LINE # 450:KD
LINE # 452:IL
LINE # 454:HP
LINE # 456:LO
LINE # 458:BD
LINE # 460:AH
LINE # 462:DA
LINE # 464:BD
LINE # 466:KH
LINE # 468:IO
LINE # 500:EM
LINE # 502:MN
LINE # 504:DB
LINE # 506:IM
LINE # 508:CF
LINE # 510:OJ
LINE # 512:DH
LINE # 514:MG
LINE # 516:FO
LINE # 518:JA
LINE # 520:DM
LINE # 522:IN
LINE # 524:AP
LINE # 526:CA
LINE # 550:CO
LINE # 552:OG
LINE # 554:KH
```

```
LINE # 556:AC
LINE # 600:LE
LINE # 602:HF
LINE # 604:LO
LINE # 606:NP
LINE # 608:MM
LINE # 610:BD
LINE # 612:KF
LINE # 614:DB
LINE # 616:GF
LINE # 618:CJ
LINE # 650:IO
LINE # 652:PD
LINE # 654:OO
LINE # 656:IN
LINE # 658:AD
LINE # 660:NP
LINE # 662:EP
LINE # 664:EP
LINE # 666:GE
LINE # 668:DM
LINE # 670:BJ
LINE # 672:HD
LINE # 674:ON
LINE # 676:FD
LINE # 678:HC
LINE # 680:JH
LINE # 682:AC
LINE # 684:GN
LINE # 686:OA
LINE # 688:IC
LINE # 700:AL
LINE # 702:BJ
LINE # 704:FC
LINE # 706:PH
LINE # 708:FG
LINE # 710:DB
LINE # 712:JJ
LINE # 714:OH
LINE # 716:BC
LINE # 718:GE
LINE # 720:EN
LINE # 722:PI
LINE # 724:OB
LINE # 726:MH
LINE # 728:OI
LINE # 730:LF
LINE # 732:NC
LINE # 734:GC
LINE # 736:OF
LINE # 738:FD
LINE # 740:BC
LINE # 742:NL
LINE # 744:PP
LINE # 746:FE
LINE # 748:AI
LINE # 750:AK
LINE # 752:CN
LINE # 754:BC
LINE # 756:LE
LINE # 758:HN
LINE # 760:PL
LINE # 762:NJ
LINE # 764:BC
LINE # 766:EK
LINE # 768:JP
LINE # 770:JK
LINE # 772:BM
LINE # 774:JK
LINE # 776:DJ
LINE # 778:IB
LINE # 780:CM
LINE # 782:EL
LINE # 784:EO
LINE # 786:HD
LINE # 788:BE
LINE # 790:JC
LINE # 792:KH
LINE # 794:OC
LINE # 796:JA
LINE # 798:HG
LINE # 800:OA
LINE # 802:II
LINE # 804:AL
LINE # 806:PG
LINE # 808:JB
LINE # 810:OE
LINE # 812:CG
LINE # 814:DM
LINE # 816:FB
LINE # 818:LB
LINE # 820:FG
LINE # 822:FJ
LINE # 824:OJ
LINE # 826:HL
LINE # 828:LO
LINE # 830:HJ
LINE # 832:BB
LINES: 234
```

SOUND EXPLORER

FROM PAGE 75

```
•100 REM * SOUND CONTROL DEMO *
•110 REM * BY DAVID BARRON    *
•120 REM * FOR AHOY! MAGAZINE *
•130 POKE53280,1:POKE53281,9
•140 PRINT"{SC}"
•150 DIMV(3,9)
•155 FM=16:CL=255:V=15:CH=7
•160 GOSUB 2300
•170 GOSUB 2700
•500 FOR X=1 TO 3
•510 FOR Y=1 TO 9
•520 GOSUB 2000
•530 PRINTV(X,Y);
•540 NEXT Y,X
•550 X=1:Y=1
•560 GOSUB 2000
•570 PRINT"{RV}";V(X,Y);
•580 GET T$
•590 IF T$="" THEN 580
•595 IF T$=" " THEN GOSUB 4000
•600 IF T$>="0" ANDT$<="9"THEN GOS
 UB 2100
•605 GOSUB 2000:PRINT"{RO}";V(X,Y)
 ;
•607 IF T$="{F7}" THEN FM=FM*2:IF
 FM=128 THEN FM=16
•610 IF T$="{CD}" THEN Y=Y+1
•620 IF T$="{CU}" THEN Y=Y-1
•630 IF T$="{CR}" THEN X=X+1
•640 IF T$="{CL}" THEN X=X-1
•642 IF T$="{F1}" THEN GOSUB 2600
•644 IF T$="{F3}" THEN GOSUB 2620
•646 IF T$="{F5}" THEN GOSUB 2640
•648 IF T$="{F4}" THEN GOSUB 2650
•650 IF X=4  THEN X=1
•660 IF X=0 THEN X=3
•670 IF Y=10 THEN Y=1
•680 IF Y=0 THEN Y=9
•685 IF T$="{F7}"THEN GOSUB 2300
•690 GOTO 560
•1999 STOP
•2000 PRINT"{HM}{CD}";
•2010 FOR X1=1 TO X*9
•2020 PRINT"{CR}";
•2030 NEXT X1
•2040 FOR Y1=1 TO Y
•2050 PRINT"{CD}";
•2060 NEXT Y1
```

IMPORTANT
Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
•2070 RETURN
•2100 GOSUB 2000
•2110 PRINT"{RO}       ";
•2120 GOSUB 2000
•2140 L$=T$
•2160 PRINTT$;
•2170 PRINT"&";
•2180 GET T$
•2185 IF T$=CHR$(13)THEN2220
•2190 IF T$<"0" OR T$>"9" THEN 218
 0
•2200 L$=L$+T$
•2205 PRINTCHR$(157);
•2210 GOTO2160
•2220 V(X,Y)=VAL(L$)
•2225 GOSUB 3000
•2230 GOSUB 2000
•2240 PRINT"{RO}      {CL}{CL}{CL}{C
 L}{RV}";V(X,Y);
•2250 RETURN
•2300 PRINT"{HM}";
•2310 PRINT"  VOICE:  1         2
     3"
•2315 PRINT
•2320 PRINT"ATTACK :"
•2330 PRINT"DECAY  :"
•2340 PRINT"SUSTAIN:"
•2345 PRINT"RELEASE:"
•2350 PRINT"HI NOTE:"
•2360 PRINT"LO NOTE:"
•2370 PRINT"P WIDTH:"
•2380 PRINT"WAVEFRM:"
•2385 PRINT"ON=1   :"
•2390 PRINT"{HM}{CD}{CD}{CD}{CD}{C
 D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}"
•2395 PRINT"VOLUME        :   {CL}{C
 L}{CL}";V
•2397 PRINT"{HM}{CD}{CD}{CD}{CD}{C
 D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
 }"
•2400 PRINT"HIGH CUTOFF :     {CL}{
 CL}{CL}{CL}";CL
•2410 PRINT"{HM}{CD}{CD}{CD}{CD}{C
 D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
 }{CD}"
•2412 PRINT"LOW CUTOFF   :     {CL}{
 CL}{CL}{CL}";CH
•2420 PRINT"{HM}{CD}{CD}{CD}{CD}{C
 D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
 }{CD}{CD}"
•2422 PRINT"RESONANCE    :   {CL}{C
 L}{CL}";RE
•2424 PRINT"{HM}{CD}{CD}{CD}{CD}{C
```

READER SERVICE INDEX

Page No.	Company	Svc. No.
35	Synapse Software	130
35	Epyx Computer Software	131
14	Commodore Software	132
14	Batteries Included	133
15	Mirage Concepts	134
39	Hayden Software	135
52-69	Protecto Enterprizes	148-165
50-51	Sight & Sound Music Software	166
12-13	Activision	167
C-2	Cardco	168
5	Mirage Concepts	169
7	Commodore	170
8	DesignWare	171
8	DesignWare	172
8	DesignWare	173
9	Sight & Sound Music Software	174
10	Futurehouse	175
37	Tech-Sketch Inc.	176
10	PSIDAC	177
37	Pronto Software	178
32	Telesys	179
9	Signal Computer Consultants	180
36	Micro-W. D.I.	181
41	Such-A-Deal	182
40	Superior Micro Systems, Inc.	183
76	M-W Dist. Inc.	184
27	Micro-Ware	185
16	Eastern House	186
38	Cadmean Corporation	187
4	PLI Micro	188
6	MSD	189
C-3	Sub Logic Corporation	190
C-4	Atari Software	191
28	Micro Technic Solutions	192
28	Computer Creations	193

```
D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
}{CD}{CD}{CD}"
•2426 PRINT"FILTER MODE :     {CL}{
CL}{CL}";
•2430 IF FM=16 THEN PRINT"LOWPASS
"
•2440 IF FM=32 THEN PRINT"BANDPASS
"
•2450 IF FM=64 THEN PRINT"HIGHPASS
"
•2470 PRINT"{CU}{CU}{CU}{CU}{CU}";
•2480 PRINTTAB(21);"RANGE:0-15   (
F1)";
•2485 PRINT"{CR}{CR}{CR}{CR}{CR}{C
R}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR
}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR}
{CR}";
•2490 PRINTTAB(21);"RANGE:0-255  (
F3)";
•2492 PRINT"{CR}{CR}{CR}{CR}{CR}{C
R}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR
}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR}
{CR}";
•2494 PRINTTAB(21);"RANGE:0-7    (
F4)";
•2495 PRINT"{CR}{CR}{CR}{CR}{CR}{C
R}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR
}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR}
{CR}";
•2500 PRINTTAB(21);"RANGE:0-15   (
F5)";
•2505 PRINT"{CR}{CR}{CR}{CR}{CR}{C
R}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR
}{CR}{CR}{CR}{CR}{CR}{CR}{CR}{CR}
{CR}";
•2510 PRINTTAB(21);"{CR}{CR}
      (F7)"
•2520 RETURN
2600 INPUT"{HM}{CD}{CD}{CD}{CD}{C
D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
}{CD}{CD}{CD}{CD}{CD}ENTER NEW VO
LUME:          {CL}{CL}{CL}{CL}{C
L}{CL}{CL}{CL}{CL}{CL}";V
•2602 IF V<0 OR V>15 THEN 2600
•2604 GOSUB 2300
•2606 GOSUB 3000
•2610 RETURN
•2620 INPUT"{HM}{CD}{CD}{CD}{CD}{C
D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
}{CD}{CD}{CD}{CD}{CD}ENTER NEW CU
TOFF LEVEL:         {CL}{CL}{CL}
{CL}{CL}{CL}{CL}{CL}{CL}{CL}";CL
•2622 IF CL<0 OR CL>255  THEN 2620
•2624 GOSUB 2300
•2626 GOSUB 3000
•2630 RETURN
•2640 INPUT"{HM}{CD}{CD}{CD}{CD}{C
D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
}{CD}{CD}{CD}{CD}{CD}ENTER NEW RE
SONANCE:            {CL}{CL}{CL}{CL
}{CL}{CL}{CL}{CL}{CL}{CL}";RE
•2642 IF RE<0 OR RE>15 THEN 2640
•2644 GOSUB 2300
•2646 GOSUB 3000
•2650 INPUT"{HM}{CD}{CD}{CD}{CD}{C
D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
}{CD}{CD}{CD}{CD}{CD}ENTER NEW CU
TOFF LEVEL:            {CL}{CL}{CL}
{CL}{CL}{CL}{CL}{CL}{CL}{CL}";CH
•2652 IF CH<0 OR CH>15 THEN 2650
•2654 GOSUB 2300
•2656 GOSUB 3000
•2660 RETURN
•2670 RETURN
•2700 REM
•2710 PRINT"{HM}{CD}{CD}{CD}{CD}{C
D}{CD}{CD}{CD}{CD}{CD}{CD}{CD}{CD
}{CD}{CD}{CD}{CD}{CD}"
•2720 PRINT"WAVEFORMS:            U
SE CURSOR CONTROL"
•2730 PRINT" 0=NOISE              T
O MOVE CURSOR."
•2740 PRINT" 1=PULSE              H
IT NUMBER TO"
•2750 PRINT" 2=SAWTOOTH           C
HANGE DATA."
•2760 PRINT" 3=TRIANGLE"
•2770 PRINT"  <SPACEBAR CONTROLS E
NVELOPE TRIGGER>";
•2780 RETURN
•3000 FOR X2=1 TO 3
•3010 FOR Y2=1 TO 9
•3020 IF V(X2,Y2)<0 THEN V(X2,Y2)=
0
•3030 NEXT Y2
•3040 FOR Y2=1 TO 4
•3050 IF V(X2,Y2)>15THEN V(X2,Y2)=
15
•3060 NEXT Y2
•3070 IF V(X2,5)>255 THEN V(X2,5)=
255
•3080 IF V(X2,6)>255 THEN V(X2,6)=
255
•3090 IF V(X2,7)>4095 THEN V(X2,7)
=4095
•3100 IF V(X2,8)>4 THEN V(X2,8)=4
```

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
3110 IF V(X2,9)>1 THEN V(X2,9)=1
3120 NEXT X2
3140 FOR X2=1 TO 3
3150 POKE 54277+(X2-1)*7,V(X2,1)*
16+V(X2,2)
3160 POKE 54278+(X2-1)*7,V(X2,3)*
16+V(X2,4)
3170 POKE 54272+(X2-1)*7,V(X2,6)
3180 POKE 54273+(X2-1)*7,V(X2,5)
3190 POKE 54275+(X2-1)*7,INT(V(X2
,7)/256)
3200 POKE 54274+(X2-1)*7,V(X2,7)-
INT(V(X2,7)/256)*256
3210 T=2↑(7-V(X2,8))
3220 T=T+(1-V(X2,9))*8
3230 POKE 54276+(X2-1)*7,T
3240 NEXT X2
3250 POKE 54296,V+FM
3260 POKE 54295,RE*16+7
3270 POKE 54293,CH
3280 POKE 54294,CL
3290 RETURN
4000 FOR X2=1 TO 3
4010 T=2↑(7-V(X2,8))
4020 T=T+(1-V(X2,9))*8
4030 POKE 54276+(X2-1)*7,T+1
4040 NEXT X2
4050 GET T$
4060 IF T$="" THEN 4050
4070 FOR X2=1 TO 3
4080 T=2↑(7-V(X2,8))
4090 T=T+(1-V(X2,9))*8
4100 POKE54276+(X2-1)*7,T
4110 NEXT X2
4120 RETURN
```

BUG REPELLENT LINE CODES FOR SOUND EXPLORER

```
LINE # 100:JF
LINE # 110:LF
LINE # 120:CJ
LINE # 130:CB
LINE # 140:HH
LINE # 150:ME
LINE # 155:IL
LINE # 160:FO
LINE # 170:FK
LINE # 500:JF
LINE # 510:KC
LINE # 520:FL
LINE # 530:FN
LINE # 540:GA
LINE # 550:LM
LINE # 560:FL
LINE # 570:PP
LINE # 580:BI
LINE # 590:FD
LINE # 595:AE
LINE # 600:NO
LINE # 605:BA
LINE # 607:FO
LINE # 610:NB
LINE # 620:HF
LINE # 630:AJ
LINE # 640:KF
LINE # 642:IG
LINE # 644:KN
LINE # 646:KA
LINE # 648:JG
LINE # 650:LG
LINE # 660:KE
LINE # 670:NK
LINE # 680:LE
LINE # 685:JC
LINE # 690:CL
LINE # 1999:JC
LINE # 2000:DE
LINE # 2010:MO
LINE # 2020:DG
LINE # 2030:AM
LINE # 2040:PI
LINE # 2050:DC
LINE # 2060:BH
LINE # 2070:IM
LINE # 2100:FL
LINE # 2110:EL
LINE # 2120:FL
LINE # 2140:KA
LINE # 2160:FB
LINE # 2170:MA
LINE # 2180:BI
LINE # 2185:JG
LINE # 2190:OF
LINE # 2200:MJ
LINE # 2205:JK
LINE # 2210:FC
LINE # 2220:NH
LINE # 2225:FE
LINE # 2230:FL
LINE # 2240:LA
LINE # 2250:IM
LINE # 2300:CE
LINE # 2310:NM
LINE # 2315:JJ
LINE # 2320:AL
LINE # 2330:MC
LINE # 2340:GB
LINE # 2345:AP
LINE # 2350:BC
LINE # 2360:BM
LINE # 2370:CL
LINE # 2380:DO
LINE # 2385:IE
LINE # 2390:LK
LINE # 2395:KP
LINE # 2397:NP
LINE # 2400:PJ
LINE # 2410:PP
LINE # 2412:CA
LINE # 2420:CG
LINE # 2422:EM
LINE # 2424:DO
LINE # 2426:AB
LINE # 2430:BB
LINE # 2440:EB
LINE # 2450:HJ
LINE # 2470:OD
LINE # 2480:CB
LINE # 2485:MN
LINE # 2490:EC
LINE # 2492:MN
LINE # 2494:AO
LINE # 2495:MN
LINE # 2500:PN
LINE # 2505:MN
LINE # 2510:IO
LINE # 2520:IM
LINE # 2600:GA
LINE # 2602:FK
LINE # 2604:FO
LINE # 2606:FE
LINE # 2610:IM
LINE # 2620:MA
LINE # 2622:CC
LINE # 2624:FO
LINE # 2626:FE
LINE # 2630:IM
LINE # 2640:HD
LINE # 2642:OB
LINE # 2644:FO
LINE # 2646:FE
LINE # 2650:LM
LINE # 2652:LE
LINE # 2654:FO
LINE # 2656:FE
LINE # 2660:IM
LINE # 2670:IM
LINE # 2700:JD
LINE # 2710:ID
LINE # 2720:KO
LINE # 2730:HL
LINE # 2740:AH
LINE # 2750:FC
LINE # 2760:KG
LINE # 2770:CA
LINE # 2780:IM
LINE # 3000:NO
LINE # 3010:NL
LINE # 3020:HN
LINE # 3030:BG
```

```
LINE # 3040:MA
LINE # 3050:NO
LINE # 3060:BG
LINE # 3070:IE
LINE # 3080:FG
LINE # 3090:NL
LINE # 3100:EO
LINE # 3110:DG
LINE # 3120:AL
LINE # 3140:NO
LINE # 3150:LM
LINE # 3160:KP
LINE # 3170:MF
LINE # 3180:JN
LINE # 3190:CO
LINE # 3200:MH
LINE # 3210:NB
LINE # 3220:KP
LINE # 3230:FF
LINE # 3240:AL
LINE # 3250:HL
LINE # 3260:GH
LINE # 3270:FJ
LINE # 3280:FC
LINE # 3290:IM
LINE # 4000:NO
LINE # 4010:NB
LINE # 4020:KP
LINE # 4030:BF
LINE # 4040:AL
LINE # 4050:BI
LINE # 4060:HA
LINE # 4070:NO
LINE # 4080:NB
LINE # 4090:KP
LINE # 4100:FF
LINE # 4110:AL
LINE # 4120:IM
LINES: 170
```

THE CASTLE OF DARKNESS

FROM PAGE 46

```
• 7 REM RD=ROOM DIRECTION TABLE.  C
 W$=COMMAND WORD TABLE.  RN$=ROOM
 NAMES
• 8 REM RC=ROOM CONTENTS TABLE.  KS
 =KEYSTROKE TABLE
• 10 DIM RD(14,8),CW$(16),RN$(60),R
 C(14),KS(64)
• 20 GOSUB 1000
• 25 GOSUB 970
• 27 REM
• 28 REM SET UP BLANK LINES AND POS
 ITION STRINGS
• 29 REM
• 30 BL$="{HM}                    {H
 M}"
• 31 RL$="{HM}{CD}{CD}":FOR I=1 TO
 78:RL$=RL$+" ":NEXT:RL$=RL$+"{HM}
 {CD}{CD}"
• 32 DL$="{HM}{CD}{CD}{CD}{CD}{CD}"
• 90 XR=1:PR=1:GOSUB 320
• 97 REM
• 98 REM MAIN LOOP.  GET KEYSTROKE
 FIRST
• 99 REM
• 100 GOSUB 990
• 105 PRINT BL$CW$(CM):IF CM=0 THEN
  100
• 110 IF CM<9 THEN GOSUB 300:GOTO 1
 00
• 115 IF CM=9 THEN GOSUB 390:GOTO 1
 00
• 180 PRINT BL$CW$(CM)
• 190 GOTO 100
• 297 REM
• 298 REM MOVEMENT HANDLER
• 299 REM
• 300 XX=XR:XR=PR:PR=RD(XR,CM)
• 307 REM
• 308 REM IS IT AN ILLEGAL MOVEMENT
 ?
• 309 REM
• 310 DS$="":IF PR>49 THEN DS$=RN$(
 PR):PR=XR:XR=XX
• 317 REM
• 318 REM SET ROOM NAME TO PRESENT
 ROOM
• 319 REM
• 320 RN$=RN$(PR)
• 327 REM
• 328 REM CLEAR SCREEN AND PRINT RO
 OM NAME AND MESSAGE
• 329 REM
• 330 PRINT "{SC}"CW$(CM)RL$RN$DL$D
 S$
• 340 RETURN
• 348 DY.
• 387 REM
• 388 REM BACK COMMAND HANDLER
• 389 REM
• 390 XX=PR:PR=XR:XR=XX:RN$=RN$(PR)
 :DS$="":GOTO 330
• 967 REM
• 968 REM RANDOMLY CHOOSE LOCATION
 OF TREASURE ROOM
• 969 REM
• 970 R=INT(10*RND(9)):IF R>2 THEN
 R=3
• 971 IF R=1 THEN R=0
• 972 ON R GOTO 975,980,985
• 975 X=2+INT(2*RND(9)):RD(10,X)=14
 :RETURN
• 980 RD(12,6)=14:RETURN
• 985 X=1+INT(8*RND(9)):R=13:IF RD(
 R,X)=55 THEN RD(R,X)=14:RETURN
• 986 GOTO 985
• 990 A=PEEK(197):IF A=64 THEN 990
• 995 CM=KS(A):RETURN
• 997 REM
• 998 REM SET UP COMMAND WORD ARRAY
• 999 REM
```

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
• 1000 FOR I=0 TO 16:READ A$:CW$(I)
=A$:NEXT
• 1010 DATA "WHAT?","NORTH","SOUTH"
,"EAST","WEST","UP","DOWN","IN","
OUT","BACK"
• 1015 DATA "TAKE","LEAVE","PEER","
GOT?","QUIT","HELP","HELP"
• 1047 REM
• 1048 REM SET UP KEYSTROKE ARRAY
• 1049 REM
• 1050 FOR I=0 TO 64:KS(I)=0:NEXT
• 1055 FOR I=1 TO 16:READ A:KS(A)=I
:NEXT
• 1060 DATA 39,13,14,9,30,18,33,38,
28,22,42,41,26,62,29,55
• 1097 REM
• 1098 REM SET UP RD (ROOM DIRECTIO
N TABLE) AND RN$ (ROOM NAME TABLE
)
• 1099 REM
• 1100 FOR I=1 TO 14:READ A$:RN$(I)
=A$:FOR J=1 TO 8:READ A:RD(I,J)=A
:NEXT:NEXT
• 1110 DATA "CASTLE MAIN GATE",52,5
3,3,2,51,50,53,50
• 1115 DATA "MEADOW WEST OF CASTLE"
,1,4,51,52,51,50,51,50
• 1120 DATA "LEDGE EAST OF CASTLE",
1,54,54,51,51,51,51,50
• 1125 DATA "GROVE SOUTH OF CASTLE"
,51,54,54,2,51,5,53,50
• 1130 DATA "KITCHEN",6,4,9,6,50,13
,9,4
• 1135 DATA "COURTYARD",7,5,9,12,7,
5,9,7
• 1140 DATA "GATEHOUSE",56,6,8,8,8,
6,6,56
• 1145 DATA "ON THE WALLS",7,10,10,
7,57,51,7,50
• 1150 DATA "GREAT HALL",6,50,50,5,
10,13,5,6
• 1155 DATA "COUNT'S CHAMBER",8,55,
55,8,11,9,50,8
• 1160 DATA "TOWER TOP",54,54,54,54
,57,10,50,50
• 1165 DATA "THE STABLES",50,50,6,5
0,50,55,50,6
• 1170 DATA "THE DUNGEONS",55,55,55
,55,9,55,50,50
• 1175 DATA "TREASURE ROOM",50,50,5
0,50,50,50,50,50
• 1197 REM
• 1198 REM SET UP RN$ VALUES FOR IL
LEGAL MOVEMENT DIRECTIONS
• 1199 REM
• 1200 FOR I=50 TO 57:READ A$:RN$(I
)=A$:NEXT
• 1210 DATA "SORRY, BUT WE CAN'T GO
 THAT WAY"
• 1215 DATA "IT'S TOO STEEP FOR US
TO CLIMB"
• 1220 DATA "WE'LL JUST GET LOST IF
 WE WANDER IN THE WOODS"
• 1225 DATA "I'M KNOCKING, BUT NOBO
DY ANSWERS"
• 1230 DATA "ARE YOU TRYING TO GET
US KILLED?"
• 1235 DATA "I SEARCHED THERE, FOUN
D NOTHING, AND    CAME BACK"
• 1240 DATA "WE DON'T HAVE WHAT IT
TAKES TO GET IN    THERE"
• 1245 DATA "UP FROM HERE?  DO YOU
THINK YOU CAN FLY?"
• 1990 RETURN
```

BUG REPELLENT LINE CODES FOR THE CASTLE OF DARKNESS

```
LINE # 7:GA          LINE # 317:JD
LINE # 8:NF          LINE # 318:DM
LINE # 10:PK         LINE # 319:JD
LINE # 20:FO         LINE # 320:CG
LINE # 25:DA         LINE # 327:JD
LINE # 27:JD         LINE # 328:DA
LINE # 28:FH         LINE # 329:JD
LINE # 29:JD         LINE # 330:OJ
LINE # 30:DO         LINE # 340:IM
LINE # 31:EJ         LINE # 348:MM
LINE # 32:AN         LINE # 387:JD
LINE # 90:NC         LINE # 388:KA
LINE # 97:JD         LINE # 389:JD
LINE # 98:GA         LINE # 390:AF
LINE # 99:JD         LINE # 967:JD
LINE # 100:DG        LINE # 968:LG
LINE # 105:KP        LINE # 969:JD
LINE # 110:DN        LINE # 970:GA
LINE # 115:AN        LINE # 971:KC
LINE # 180:PL        LINE # 972:JG
LINE # 190:CF        LINE # 975:HF
LINE # 297:JD        LINE # 980:IB
LINE # 298:AF        LINE # 985:KE
LINE # 299:JD        LINE # 986:DI
LINE # 300:EC        LINE # 990:EM
LINE # 307:JD        LINE # 995:EG
LINE # 308:CO        LINE # 997:JD
LINE # 309:JD        LINE # 998:PI
LINE # 310:IN        LINE # 999:JD
```

```
LINE # 1000:EL      LINE # 1150:IF
LINE # 1010:FK      LINE # 1155:JP
LINE # 1015:IB      LINE # 1160:BK
LINE # 1047:JD      LINE # 1165:DF
LINE # 1048:HD      LINE # 1170:AD
LINE # 1049:JD      LINE # 1175:AD
LINE # 1050:NI      LINE # 1197:JD
LINE # 1055:AI      LINE # 1198:LF
LINE # 1060:PJ      LINE # 1199:JD
LINE # 1097:JD      LINE # 1200:DH
LINE # 1098:NH      LINE # 1210:IM
LINE # 1099:JD      LINE # 1215:PK
LINE # 1100:JI      LINE # 1220:JI
LINE # 1110:PP      LINE # 1225:KG
LINE # 1115:PH      LINE # 1230:KF
LINE # 1120:IP      LINE # 1235:HB
LINE # 1125:KB      LINE # 1240:AC
LINE # 1130:MJ      LINE # 1245:GM
LINE # 1135:GM      LINE # 1990:IM
LINE # 1140:MK      LINES: 98
LINE # 1145:CJ
```

BASE CONVERSIONS

FROM PAGE 77

```
1 REM*****************************
******
2 REM*THIS PROGRAM WILL EXCHANGE
BINARY
3 REM*HEXADECIMAL AND DECIMAL NOT
ATION
4 REM* LINES 10 - 80 DISPLAY THE
MENU           * AND GET THE FUNCTIO
N NUMBER
5 REM*****************************
******
6 POKE53281,7:N$="":E$=""
7 PRINT"{SC}{CD}{CD}{CD}{CD}{CD}{
RV}"
8 PRINT E$
10 PRINT"{RV}{RD}CHOSE A FUNCTION
 1-7{RO}{BL}"
20 PRINT"1 BINARY TO DECIMAL"
30 PRINT"2 BINARY TO HEXADECIMAL"
40 PRINT"3 HEXADECIMAL TO DECIMAL
"
50 PRINT"4 HEXADECIMAL TO BINARY"
60 PRINT"5 DECIMAL TO BINARY"
70 PRINT"6 DECIMAL TO HEXADECIMAL
"
75 PRINT"7 EXIT":INPUT A:IF(A>7)O
R(A<1)THENE$="NONEXISTANT FUNCTIO
N":GOTO 7
77 IF A=7 THEN END
80 PRINT"ENTER YOUR NUMBER":E$=""
81 REM****************************
******        * THE NEXT FEW LINES
 ACCEPT THE
82 REM* VALUE INTO A STRING VARIA
BLE            * AND SEND IT TO THE
 SUBROUTINES
83 REM* FOR CONVERSION
            *********************
*************
90 INPUT N$:ON A GOSUB 1000,200,1
200,400,1300,1100
95 PRINT"{SC}{RV}{RD}{CD}{CD}{CD}
{CD}{CD}{CR}{CR}THE ANSWER IS "N$
:PRINT:PRINT"{GN}PRESS ANY KEY TO
 GO TO MENU"
96 GETXZ$:IFXZ$=""THEN GOTO96
97 GOTO 7
200 GOSUB 1000
210 GOSUB 1100
220 RETURN
400 GOSUB 1200
410 GOSUB 1300
420 RETURN
1000 REM*************************
*****
1005 REM*THIS SUBROUTINE CONVERTS
1010 REM* BINARY TO DECIMAL
1015 REM*************************
****
1020 L=LEN(N$):DN=0:D$=""
1030 IF L>16 THEN E$="DATA TOO LA
RGE":GOTO 7
1035 FORX=1 TO L
1040 IFNOT(MID$(N$,X,1)="0"ORMID$
(N$,X,1)="1")THEN E$="BAD BIN DAT
A":GOTO 7
1045 NEXT
1050 FOR X=LTO1 STEP-1
1055 CH$=MID$(N$,X,1)
1060 CH=0
1065 IF CH$="1" THEN CH=1
1070 DN =DN+(2↑(L-X)*CH)
1075 NEXT X
1080 N$=STR$(DN):FORX=1 TO LEN(N$
)
1085 IF NOT MID$(N$,X,1)=" "THEN
D$=D$+MID$(N$,X,1)
1090 NEXT:N$=D$:RETURN
1100 REM*************************
```

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
*******
•1101 REM*THIS SUBROUTINE WILL CON
VERT
•1102 REM*DECIMAL TO HEXADECIMAL
•1103 REM*************************
*****
•1104 L=LEN(N$)
•1105 FOR X=1 TO L:C$=MID$(N$,X,1)
•1110 IFASC(C$)<48ORASC(C$)>57THEN
E$="ILLEGAL DECIMAL DATA":GOTO 7
•1115 NEXT X:IF VAL(N$)>65535THEN
E$="NUMBER TOO LARGE":GOTO 7
•1120 D=VAL(N$):ND=1:N$=""
•1125 IF 16↑ND>=D+1THEN GOTO 1135
•1130 ND=ND+1:GOTO 1125
•1135 FOR X=ND TO 1 STEP-1
•1145 C=INT(D/16↑(X-1))
•1150 D=D-C*16↑(X-1)
•1155 IF C=15 THEN N$=N$+"F"
•1160 IF C=14 THEN N$=N$+"E"
•1165 IF C=13 THEN N$=N$+"D"
•1170 IF C=12 THEN N$=N$+"C"
•1175 IF C=11 THEN N$=N$+"B"
•1180 IF C=10 THEN N$=N$+"A"
•1185 IF C<10 THEN N$=N$+STR$(C)
•1190 NEXT X :D$="":FOR X=1 TO LEN
(N$)
•1195 IFNOTMID$(N$,X,1)=" "THEND$=
D$+MID$(N$,X,1)
•1196 NEXT:N$=D$:RETURN
•1200 REM*************************
******
•1201 REM* THIS SUBROUTINE WILL CO
NVERT          * HEXADECIMAL TO D
ECIMAL
•1202 REM*************************
******
•1204 FORX=1 TO LEN(N$):D$=MID$(N$
,X,1)
•1205 IFASC(D$)<48 OR ASC(D$)>70TH
ENE$="ILLEGAL HEX DATA":GOTO 7
•1206 IFASC(D$)<65ANDASC(D$)>57 TH
EN E$="ILLEGAL HEXADECIMAL DATA"
GOTO 7
•1207 IF LEN(N$)>4 THEN E$="NUMBER
 TOO LARGE":GOTO 7
•1208 NEXT
•1209 N=0:D$=""
•1210 FOR X=0 TO LEN(N$)-1
•1215 CH$=MID$(N$,LEN(N$)-X,1)
•1220 IF CH$="F" THEN F=15
•1225 IF CH$="E" THEN F=14
•1230 IF CH$="D" THEN F=13
•1235 IF CH$="C" THEN F=12
•1240 IF CH$="B" THEN F=11
•1245 IF CH$="A" THEN F=10
•1250 IF ASC(CH$)<58 THEN F=VAL(CH
$)
•1255 N=N+F*16↑X
•1260 NEXT X
•1265 N$=STR$(N)
•1270 FOR X=1 TO LEN(N$)
•1280 IFNOT MID$(N$,X,1)=" "THEN D
$=D$+MID$(N$,X,1)
•1285 NEXT:N$=D$:RETURN
•1299 REM*************************
*******        * THIS SUBROUTINE
WILL CONVERT
•1300 REM* DECIMAL TO BINARY
                 ******************
*************
•1301 FORX=1 TO LEN(N$)
•1302 IFASC(MID$(N$,X,1))>57ORASC(
MID$(N$,X,1))<48THEN E$="BAD DEC
DATA":GOTO7
•1303 NEXT:IFVAL(N$)>65535THEN E$=
"NUMBER TOO LARGE":GOTO 7
•1309 BC=0:C=VAL(N$):N$=""
•1310 IF 2↑BC<C THEN BC=BC+1:GOTO
1310
•1315 FOR X=0 TO BC
•1320 C$="1"
•1325 IF C<2↑BC THEN C$="0"
•1330 IF C>=2↑BC THEN C=C-2^BC
•1335 BC=BC-1
•1340 N$=N$+C$
•1345 NEXT X
•1350 RETURN
```

BUG REPELLENT LINE CODES FOR BASE CONVERSIONS (C-64)

```
LINE #   1:EO        LINE #  75:BA
LINE #   2:FI        LINE #  77:OC
LINE #   3:MN        LINE #  80:HA
LINE #   4:FP        LINE #  81:DE
LINE #   5:EO        LINE #  82:CK
LINE #   6:FH        LINE #  83:DJ
LINE #   7:NL        LINE #  90:AC
LINE #   8:AP        LINE #  95:MD
LINE #  10:LB        LINE #  96:LM
LINE #  20:NI        LINE #  97:MC
LINE #  30:PL        LINE # 200:FO
LINE #  40:DF        LINE # 210:FH
LINE #  50:HB        LINE # 220:IM
LINE #  60:AA        LINE # 400:FA
LINE #  70:OO        LINE # 410:FB
```

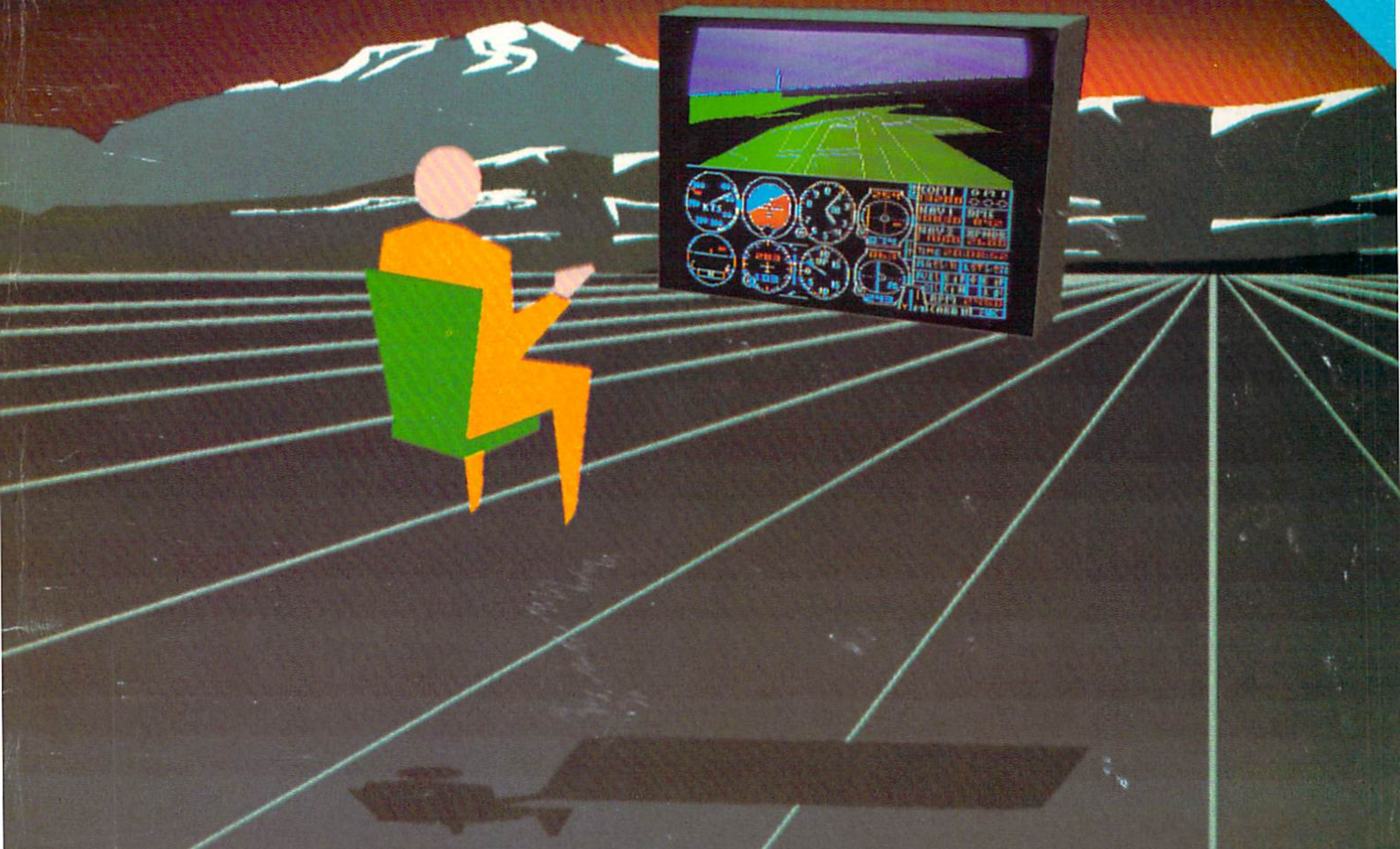
```
LINE # 420:IM
LINE # 1000:NK
LINE # 1005:EG
LINE # 1010:PC
LINE # 1015:KF
LINE # 1020:DC
LINE # 1030:PD
LINE # 1035:KM
LINE # 1040:FK
LINE # 1045:IA
LINE # 1050:DB
LINE # 1055:MP
LINE # 1060:GM
LINE # 1065:IG
LINE # 1070:MA
LINE # 1075:NK
LINE # 1080:EE
LINE # 1085:AE
LINE # 1090:CO
LINE # 1100:DD
LINE # 1101:FJ
LINE # 1102:IE
LINE # 1103:NK
LINE # 1104:JP
LINE # 1105:KA
LINE # 1110:NN
LINE # 1115:EB
LINE # 1120:ML
LINE # 1125:MG
LINE # 1130:EP
LINE # 1135:IL
LINE # 1145:LH
LINE # 1150:FC
LINE # 1155:HO
LINE # 1160:GO
LINE # 1165:JC
LINE # 1170:HC
LINE # 1175:GO
LINE # 1180:FO
LINE # 1185:DM
LINE # 1190:GL
LINE # 1195:AE
LINE # 1196:CO
LINE # 1200:CE
LINE # 1201:MM
LINE # 1202:CE
LINE # 1204:LD
LINE # 1205:FC
LINE # 1206:KE
LINE # 1207:HK
LINE # 1208:IA
LINE # 1209:OO
LINE # 1210:NO
LINE # 1215:DN
LINE # 1220:KF
LINE # 1225:JJ
LINE # 1230:JN
LINE # 1235:KJ
LINE # 1240:IN
LINE # 1245:IB
LINE # 1250:PA
LINE # 1255:FF
LINE # 1260:NK
LINE # 1265:IA
LINE # 1270:NK
LINE # 1280:AE
LINE # 1285:CO
LINE # 1299:DJ
LINE # 1300:ID
LINE # 1301:NK
LINE # 1302:OM
LINE # 1303:KJ
LINE # 1309:LI
LINE # 1310:HK
LINE # 1315:OB
LINE # 1320:JG
LINE # 1325:DA
LINE # 1330:KL
LINE # 1335:IF
LINE # 1340:JM
LINE # 1345:NK
LINE # 1350:IM
LINES: 112
```

BUG REPELLENT LINE CODES FOR BASE CONVERSIONS (VIC 20)

```
LINE # 1:EO
LINE # 2:PN
LINE # 3:OD
LINE # 4:PK
LINE # 5:EO
LINE # 6:FH
LINE # 7:NL
LINE # 8:AG
LINE # 10:KE
LINE # 20:KB
LINE # 30:GG
LINE # 40:OA
LINE # 50:IK
LINE # 60:HL
LINE # 70:ND
LINE # 75:DN
LINE # 77:NJ
LINE # 80:OL
LINE # 81:MI
LINE # 82:KH
LINE # 83:EO
LINE # 90:PJ
LINE # 95:JE
LINE # 96:MM
LINE # 97:ME
LINE # 200:FP
LINE # 210:FE
LINE # 220:IM
LINE # 400:FJ
LINE # 410:FG
LINE # 420:IM
LINE # 1000:NK
LINE # 1005:MM
LINE # 1010:CE
LINE # 1015:KF
LINE # 1020:DC
LINE # 1030:HC
LINE # 1035:LB
LINE # 1040:KF
LINE # 1045:IA
LINE # 1050:EI
LINE # 1055:MP
LINE # 1060:GM
LINE # 1065:GK
LINE # 1070:JD
LINE # 1075:NM
LINE # 1080:EH
LINE # 1085:LL
LINE # 1090:CO
LINE # 1100:DD
LINE # 1101:FJ
LINE # 1102:IO
LINE # 1103:NK
LINE # 1104:JP
LINE # 1105:EK
LINE # 1110:JP
LINE # 1115:KJ
LINE # 1120:ML
LINE # 1125:MG
LINE # 1130:EO
LINE # 1135:II
LINE # 1145:LH
LINE # 1150:FC
LINE # 1155:GI
LINE # 1160:GI
LINE # 1165:HI
LINE # 1170:EI
LINE # 1175:EI
LINE # 1180:DI
LINE # 1185:CD
LINE # 1190:KD
LINE # 1195:NG
LINE # 1196:CO
LINE # 1200:CE
LINE # 1201:DP
LINE # 1202:CE
LINE # 1204:LO
LINE # 1205:FP
LINE # 1206:LO
LINE # 1207:CO
LINE # 1208:IA
LINE # 1209:OO
LINE # 1210:MJ
LINE # 1215:DN
LINE # 1220:IP
LINE # 1225:LF
LINE # 1230:IH
LINE # 1235:KF
LINE # 1240:JP
LINE # 1245:GF
LINE # 1250:NE
LINE # 1255:FF
LINE # 1260:NM
LINE # 1265:IA
LINE # 1270:OP
LINE # 1280:PN
LINE # 1285:CO
LINE # 1299:CF
LINE # 1300:AC
LINE # 1301:OP
LINE # 1302:BD
LINE # 1303:AM
LINE # 1309:LI
LINE # 1310:MJ
LINE # 1315:OP
LINE # 1320:JG
LINE # 1325:OP
LINE # 1330:FN
LINE # 1335:IF
LINE # 1340:JM
LINE # 1345:NM
LINE # 1350:IM
LINES: 112
```

Flight Simulator II

For the Commodore 64™



Put yourself in the pilot's seat of a Piper 181 Cherokee Archer for an awe-inspiring flight over realistic scenery from New York to Los Angeles. High speed color-filled 3D graphics will give you a beautiful panoramic view as you practice takeoffs, landings, and aerobatics. Complete documentation will get you airborne quickly even if you've never flown before. When you think you're ready, you can play the World War I Ace aerial battle game. Flight Simulator II features include ■ animated color 3D graphics ■ day, dusk, and night flying modes ■ over 80 airports in four scenery areas: New York, Chicago, Los Angeles, Seattle, with additional scenery areas available ■ user-variable weather, from clear blue skies to grey cloudy conditions ■ complete flight instrumentation ■ VOR, ILS, ADF, and DME radio equipped ■ navigation facilities and course plotting ■ World War I Ace aerial battle game ■ complete information manual and flight handbook.

See your dealer . . .

or for direct orders enclose $49.95 plus $2.00 for shipping and specify UPS or first class mail delivery. American Express, Diner's Club, MasterCard, and Visa accepted.

Order Line: 800 / 637-4983

subLOGIC Corporation
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex: 206995

Reader Service No. 190

ATARISOFT. All the hits your computer is missing.

It's showtime.

Time for ATARISOFT™ to show you six exciting, brand new games that are destined for stardom.

Games that can be played on your Commodore 64, IBM PC and Apple II. (Some titles available on IBM PC jr. and VIC 20.*)

First, there's *Gremlins*[1], based on the characters from the original film presented by Steven Spielberg.

Then there's *Crystal Castles*™ where Bentley Bear™ journeys through all sorts of tantalizingly difficult paths and ramps in his endless quest for gems.

In *Donkey Kong Jr.*[2] by Nintendo[2], Junior tries to rescue his father against immense odds. And speaking of Donkey Kong, there's also *Mario Brothers*[2] by Nintendo[2]. This time, Mario and his brother Luigi battle creatures on four levels of floors, encountering all sorts of treacherous enemies.

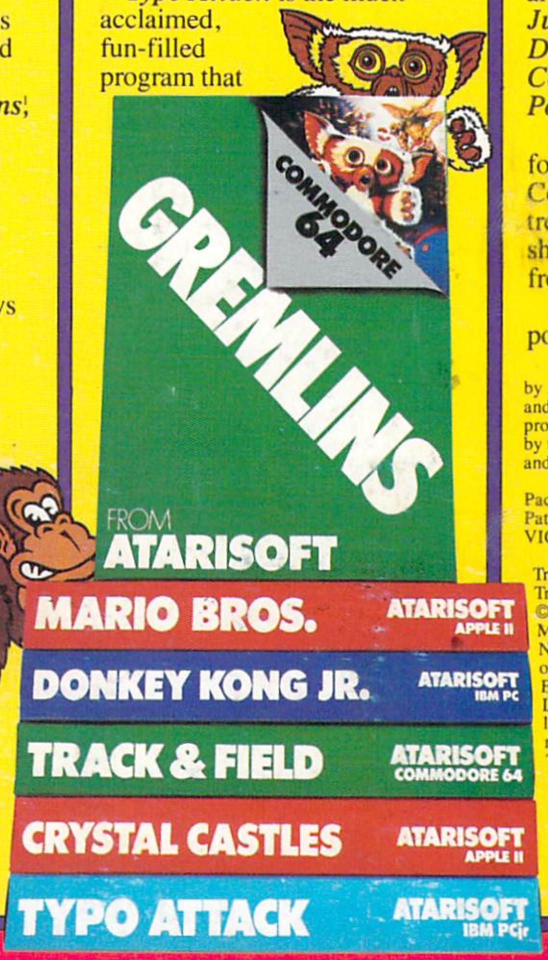
In *Track And Field*[3] you can compete by yourself or head-to-head with another player. But each player must beat qualifying times, heights and distances before they can compete in each of the grueling six events.

Typo Attack is the much-acclaimed, fun-filled program that allows you to enjoy developing your typing skills at any level.

And still playing to the delight of audiences everywhere are *Pac-Man*[4], *Ms. Pac-Man*[4], *Jungle Hunt*[5], *Battlezone*™, *Donkey Kong*[2] by Nintendo[2], *Centipede*™ and *Pole Position*[6].

So, if you've been searching for ways to entertain your Commodore, Apple or IBM, treat it to one of the best shows in town, one of the hits from ATARISOFT.

And don't forget the popcorn.



ATARISOFT products are manufactured by Atari, Inc. for use with various computers and video game consoles. ATARISOFT products are not made, licensed or approved by the manufacturer(s) of those computers and video game consoles.

*Titles available on IBM PC jr. are Ms. Pac-Man, Centipede, Donkey Kong, Moon Patrol[7] and Typo Attack. Available on the VIC 20 is Typo Attack.

1. © 1984 Warner Brothers, Inc.; 2. Trademarks and © Nintendo, 1982, 1983; 3. Trademark of Konami Industry Co., Ltd., © 1983 of Konami; 4. Trademarks of Bally Mfg. Co. Sublicensed to Atari, Inc. by Namco-America, Inc.; 5. Trademark and © of Taito America Corporation 1982; 6. Engineered and designed by Namco Ltd., manufactured under license by Atari, Inc. Trademark and © Namco 1982; 7. Trademark and © Williams 1982, manufactured under license from Williams Electronics, Inc.

ATARISOFT

SIX NEW HITS ARE COMING SOON TO A COMMODORE 64 SCREEN NEAR YOU.

Reader Service No. 191